



ROYAUME DU MAROC
UNIVERSITE CHOUAIB DOUKKALI
FACULTE DES SCIENCES EL JADIDA
DEPARTEMENT D'INFORMATIQUE



Projet de Fin d'étude

Pour l'obtention du Diplôme Licence Professionnelle

Administrateur de Base de Données

Sous le thème

Application et API
« Gestion du personnel »

Présenté par : GUENNOUN Ismail

Soutenu le 20 Octobre 2022 devant les membres de jury :

Pr. Mohammed Essaid RIFFI : Faculté des Sciences, El Jadida

Encadrant

Pr. Abdellah MADANI : Faculté des Sciences, El Jadida

Examineur

Année universitaire 2021/2022

Remerciements

La réalisation de ce mémoire a été possible grâce au concours de plusieurs personnes à qui je voudrais témoigner toute ma gratitude.

Mes très chers parents, pour leur soutien inconditionnel et leurs encouragements incessant.

Mme « **T. Amal** » qui m'a trouvée l'opportunité de continuer mes études.

Mon ami qui voudrait rester sous le nom anonyme de « **LOUISAMBE** » ainsi que mon ami « **S.T. Yassine** », pour leurs soutiens logistique et intellectuel.

L'équipe pédagogique des professeurs et enseignants chargé de la licence professionnel « Administration de Bases de Données » pour sa patience, disponibilité et conseils judicieux, qui m'a permis d'acquérir de nouvelles connaissances théoriques et pratiques, sous les fines directives du « **Pr. RIFFI Mohammed Essaid** » responsable de la filière et mon encadrant pour ce projet de mémoire.

Le personnel de la Direction Régional des Impôts de Marrakech qui de près ou de loin ont contribué à la réalisation de ce projet de fin d'études, chapoté par « **Mr. ATTAOUI Bennaceur** » chef du Service Régional des Ressources et du Système d'Information, et « **Mr. OULYALI Mouhcine** » analyste programmeur qui n'a cessé d'alimenter ma réflexion et de m'apporter ses conseils techniques résultantes de plusieurs années d'expérience.

ملخص

إنَّ الحمد لله نحمده ونستعينه ونستغفره، ونعوذ بالله من شرور أنفسنا وسيئات أعمالنا، مَنْ يَهْدِهِ اللهُ فلا مُضِلَّ لَهُ، وَمَنْ يُضِلِلْ فلا هادي لَهُ، وأشهد أن لا إله إلا الله وحده لا شريك لَهُ، وأشهد أنَّ محمدًا عبده ورسوله، صَلَّى اللهُ وسلَّمَ وبارك عليه، وعلى آله وصحبه أجمعين.

أما بعد:

هذه الوثيقة هي خلاصة العمل المنجز في إطار مشروع تدريجي خلال فترة تدريبي في المديرية الجهوية للضرائب بمراكش. الهدف من هذا المشروع هو تصميم وبرمجة (1) API لـ "إدارة شؤون الموظفين" بالإضافة إلى برنامج واجهة للمستخدم النهائي.

النمو المتزايد في عدد الموظفين داخل المديرية الجهوية للضرائب بمراكش وكذلك التعيينات وإعادة التعيينات وفقًا لاستراتيجية إعادة توزيع الموارد البشرية؛ لم تعد تسمح بتدبير هذه البيانات ومراقبتها اعتمادًا فقط على ملف « Excel » ، خاصة وأن هذه المهمة لم تعد تحت إشراف شخص واحد فقط، ولا تسمح بمشاركة هذه البيانات أنيا مع المركز الجهوي للمعلومات، لتدبير الأجهزة الرقمية ناهيك عن فقدان أرشيف البيانات.

أدت الدراسة المتعمقة لملف « Excel » باستخراج قاعدة بيانات (بنيتا و معلومات) التي سيتم التواصل معها فقط بواسطة ال API أي الخادم الخلفي (2)(Back-end)، لتوفير التفاعل المطلوب لواجهة المستخدم (Font-end) أو أي نظام خارجي متاح (أنظمة الطرف الثالث)، مع الأخذ بعين الاعتبار التقنيات الموافقة للبيئة التقنية الموجودة في الإدارة.

Résumé

Le présent document est la synthèse de mon travail accompli dans le cadre du projet de fin d'études au sein de la Direction Régional des Impôts de Marrakech. L'objectif de ce projet et la conception et la réalisation d'une API de « Gestion du personnel » avec une application web d'utilisation.

La croissance du nombre du personnel au sein de la DRI de Marrakech ainsi que les affectations et les réaffectations permanentes selon la stratégie de redéploiement des ressources humaines; ne permet plus de faire la gestion et le suivi de ces données en se basant sur un fichier Excel, surtout que cette tâche n'est plus dédiée à une seule personne et ne permet pas la synchronisation en temps réel des données avec le CRI pour la gestion du matériel sans parler de la perte de l'historique.

L'étude approfondie du « fichier Excel » a permis l'extraction de la base de données qui sera interrogée uniquement par l'API ⁽¹⁾ (Back-end) ⁽²⁾, pour fournir l'interaction souhaitée à l'interface utilisateur (Font-end) ou des tiers systèmes, tout en utilisant les technologies compatibles avec l'environnement technique disponible.

Abstract

This document is the synthesis of my work accomplished within the graduation project during my internship in the Regional Tax Department of Marrakesh. The objective of this project is to design and implement an API for "Personnel Management" in addition to an end-user application.

The growth in the personnel number within RTD of Marrakesh as well as the permanent assignments and reassignments according to the human resources redeployment strategy; do no longer allows the management of these data solely based on an Excel file, especially since this task is no longer dedicated to a single person and does not allow real-time data synchronization with the IT department for hardware management purposes, not to mention the no history log.

The in-depth study of the "Excel file" allowed the extraction of the database, which would be queried only by the API⁽¹⁾ (Back-end)⁽²⁾, to provide the required interaction to the user interface (Font-end) or third-party systems, while using compatible technologies with the available technical environment.

Table des matières

Remerciements.....	i
ملخص.....	ii
Résumé.....	iii
Abstract.....	iv
Table des matières	v
Introduction Générale	1
Chapitre I : Présentation Générale	3
I.1 Introduction	3
I.2 Présentation de l'organisme d'accueil.....	3
I.2.1 Présentation de la DRI	3
I.2.2 Organigramme des Services de la DRI Marrakech.....	4
I.2.3 Valeurs	4
I.2.4 Missions	5
I.2.5 Vision stratégique	6
I.3 Présentation du projet.....	7
I.3.1 Contexte du projet.....	7
I.3.2 Objectifs du projet.....	7
I.4 Conclusion.....	7
Chapitre II : Gestion de Projet	8
II.1 Introduction	8
II.2 Méthodologie	8
II.3 Approche du projet.....	9
II.4 Planification et suivi.....	10

II.5	Planning.....	11
II.6	Conclusion.....	11
Chapitre III : Etude de l'existant		12
III.1	Introduction	12
III.2	Méthode de gestion utilisée	12
III.3	Problématique	13
III.4	Environnement technique disponible	13
III.5	Etude comparative	14
III.6	Conclusion	15
Chapitre IV : Analyse et conception.....		16
IV.1	Introduction	16
IV.2	Analyse des besoins.....	16
IV.2.1	Identification des acteurs	16
IV.2.2	Spécification des besoins fonctionnels	17
IV.2.3	Spécification des besoins non fonctionnels	18
IV.3	Conception.....	19
IV.3.1	Diagrammes de cas d'utilisation.....	20
IV.3.2	Diagrammes de séquences	21
IV.3.3	Diagramme de classes.....	24
IV.4	Conclusion	25
Chapitre V : Réalisation		26
V.1	Introduction	26
V.2	Architecture, technologies et outils de développement.....	26
V.2.1	Architecture.....	26
V.2.2	Technologies	27
V.2.3	Outils de développement.....	29
V.3	API (Back-end)	30
V.3.1	Arborescence.....	30

V.3.2	Captures d'écrans.....	31
V.3.3	Exemples de code	33
V.4	Application Web (Front-End)	36
V.4.1	Arborescence.....	36
V.4.2	Captures d'écrans.....	36
V.4.3	Exemples de code	42
V.5	Conclusion.....	45
Conclusion Générale.....		46
Annexe A Colonnes du fichier « Excel » du traitement initial.....		47
Liste des figures		48
Liste des tableaux.....		49
Liste des abréviations.....		50
Glossaire des termes techniques		51
Webographie		55

Introduction Générale

Dans le cadre de l'amélioration de la qualité de gestion des ressources humaines au sein de la Direction Régionale des Impôts de Marrakech pour fournir de meilleurs services et prestations aux contribuables, un bon déploiement de ces ressources est primordial surtout que la structure de l'administration change constamment et que les employés acquièrent de nouvelles connaissances et diplômes. Donc il faut mettre la bonne compétence au bon post.

C'est dans cette optique, que le bureau des ressources humaines essaye de rassembler et de traiter toutes les données nécessaires à la prise de décision, afin de garantir des affectations optimales aux postes vacants et de responsabilités en prenant en considération les villes des bâtiments annexes de la DRI de Marrakech.

Cette gestion se faisait jusqu'à maintenant par une personne depuis un fichier Excel. Qui suite à l'augmentation de l'effective du bureau RH ne permet plus le traitement simultané sur le même fichier, ainsi que les historiques antérieurs (Adresse, Poste, Unité de gestion ... etc.).

De plus que la dernière situation devrait toujours être partagé avec le centre régional d'informatique pour des raisons d'affectation d'équipements informatique, ainsi que le service central des RH relevant de la Direction général des Impôts qui centralise les données du personnel au niveau du ministère des finances.

Pour ces raisons, il est indispensable de s'offrir les outils permettant de subvenir aux besoins déjà cités. C'est dans ce cadre que vient la solution développée dans ce projet.

Le présent rapport synthétise les différentes étapes de réalisation organisées comme suit :

Le premier chapitre est dédié à la présentation de l'organisme d'accueil « DRI de Marrakech » et la présentation du projet.

Le deuxième chapitre sera consacré à la gestion du projet (Approche, méthodologie et planification).

Le troisième chapitre décrira la problématique ainsi qu'une étude de l'environnement technique disponible, et une comparative des solutions présente sur le marché.

Le quatrième chapitre s'intéressera à l'analyse et la conception du projet.

Le cinquième chapitre présentera la réalisation du projet ainsi que les technologies adoptées.

Finalement nous clôturant ce rapport par une conclusion globale du travail réalisé avec certaines perspectives d'amélioration possibles.

Chapitre I :

Présentation Générale

I.1 Introduction

Ce chapitre sera consacré dans un premier temps à la présentation de l'organisme d'accueil, son organigramme et dans un deuxième temps à la présentation du projet.

I.2 Présentation de l'organisme d'accueil

I.2.1 Présentation de la DRI

La Direction Régional des Impôts de Marrakech est l'une des 9 direction régionaux des impôts du Maroc. Sa mission principale consiste à assurer les recettes fiscales pour remplir les caisses de l'état. Régie par la Direction Générale des Impôts ⁽³⁾ qui se charge de la mise en œuvre de la politique fiscale et des relations entre l'administration et les usagers, conformément aux orientations du Ministère de l'Economie et des Finances.

Ses services sont spécialisés pour gérer distinctement les dossiers des particuliers, des professionnels, des personnes morales et des grandes entreprises, dans le but de fournir un service de proximité adapté à chaque profil. Chaque contribuable s'adresse ainsi à un interlocuteur unique polyvalent qui traite son dossier au titre de ses différents impôts.

Globalement, les Directions Régionales, Inter Préfectorales et Préfectorales sont organisées autour de trois fonctions principales ⁽⁴⁾:

- L'assiette de l'impôt qui est gérée au niveau des subdivisions d'assiette spécialisées par type de contribuables.
- Le recouvrement des impôts effectué au niveau des recettes de l'administration fiscale.
- Le contrôle fiscal accompli par les brigades de vérification.

I.2.2 Organigramme des Services de la DRI Marrakech

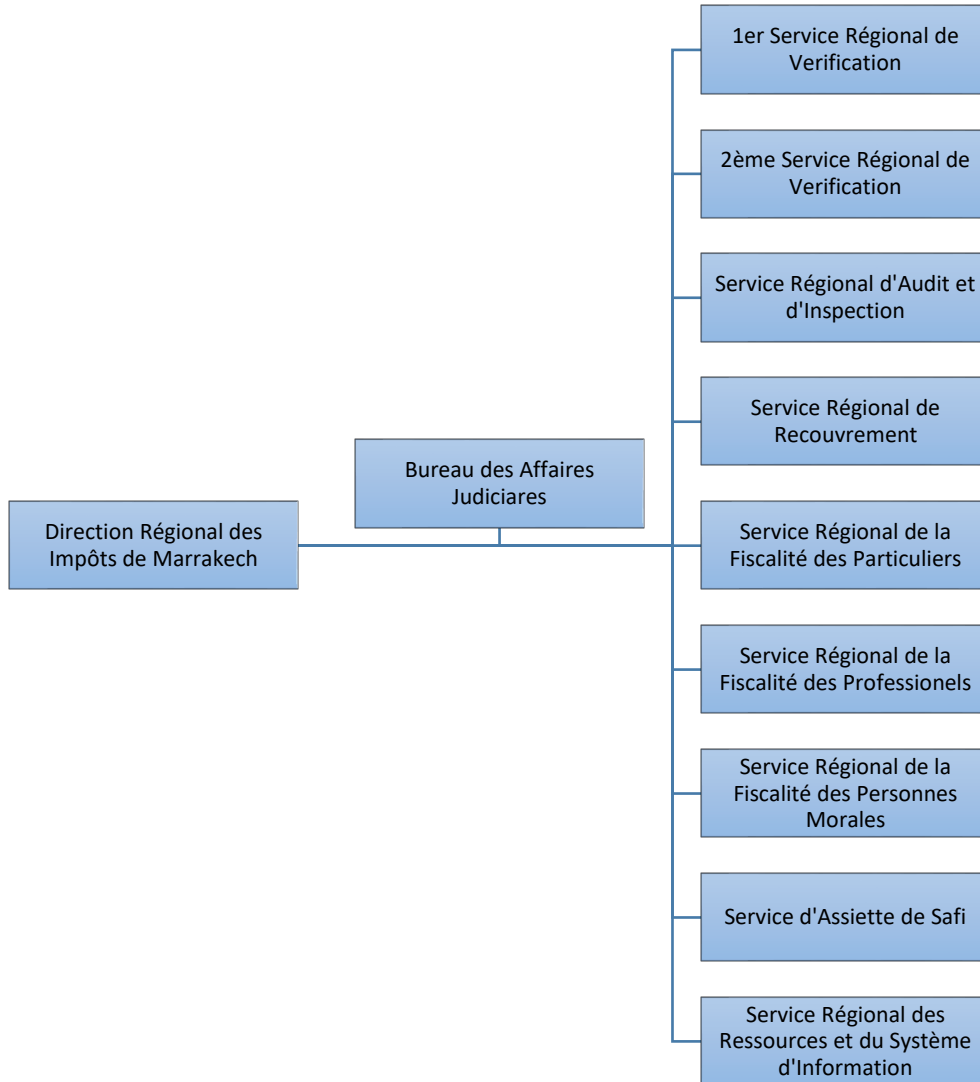


Figure 1 : Organigramme de la DRI de Marrakech

I.2.3 Valeurs ⁽⁵⁾

Cinq valeurs essentielles sont portées et partagées par les fonctionnaires de la DGI, dans le cadre de l'exercice quotidien de leurs fonctions et de leurs relations avec les usagers.

INTEGRITE : Rigueur, probité, transparence, éthique, discrétion.

PROFESSIONNALISME : Efficacité, fiabilité, célérité, assistance, service, réactivité, proximité.

RESPECT : Disponibilité, écoute, courtoisie, considération

NEUTRALITE : Equité

INNOVATION : Ouverture, créativité, initiative

I.2.4 Missions⁽⁶⁾

Asseoir et recouvrer les impôts d'état :

- Les impôts d'Etat (impôt sur les sociétés, impôt sur le revenu, taxe sur la valeur ajoutée, droits d'enregistrement et de timbre) ;
- Certains impôts locaux gérés pour le compte des collectivités locales (taxe d'habitation, taxe des services communaux et taxe professionnelle).

Les recettes de l'administration fiscales se chargent de l'encaissement des recettes des impôts, droits et taxes.

S'assurer du respect de la loi fiscale et lutter contre la fraude fiscale :

Le système fiscal marocain est globalement déclaratif. La mission de contrôle occupe donc une dimension très importante.

Le contrôle fiscal s'opère dans le cadre d'une procédure stricte observant les garanties offertes aux assujettis. Il a un rôle dissuasif et pédagogique.

Fournir au gouvernement une expertise fiscale notamment dans le cadre de l'élaboration des lois de finances et de la négociation des conventions de non double imposition avec les états étrangers :

La Direction Générale des Impôts est ainsi chargée d'étudier et d'élaborer les projets de textes législatifs et réglementaires à caractère fiscal.

Elle élabore également, en concertation avec les différents opérateurs économiques, les circulaires d'application relatives aux textes fiscaux.

L'administration fiscale étudie en outre les questions d'interprétation des dispositions fiscales et informe sur les réponses apportées.

I.2.5 Vision stratégique ⁽⁷⁾

La Direction Générale des Impôts s'inscrit dans une stratégie de performance et d'innovation pour assurer la mobilisation des ressources avec équité et garantir un service de qualité au citoyen.

La Vision stratégique de la DGI exprime la détermination des femmes et des hommes qui la composent, à relever de nouveaux défis. Elle dresse la feuille de route qui guide leurs priorités et leurs actions.

Elle consolide leur vocation de service au citoyen et les recentre sur les impératifs de performance dans la collecte des ressources budgétaires.

La vision stratégique de la DGI repose sur quatre orientations :

Construire une relation de partenariat et de confiance avec le contribuable :

- Faciliter les différentes démarches et procédures liées au contribuable ;
- Garantir au contribuable ses droits ;
- Communiquer et favoriser l'adhésion à l'Impôt et le civisme fiscal ;
- Renforcer l'éthique.

Devenir une administration numérique :

Développer et généraliser des télé-procédures sécurisées.

Renforcer l'action de l'administration en matière de contrôle :

- Disposer d'informations pertinentes, fiables et exploitables ;
- Mettre en place les ressources et outils nécessaires à l'analyse des risques et au ciblage de la fraude.

Développer une parfaite maîtrise du métier du recouvrement :

- Renforcer les structures et les ressources dédiées au recouvrement ;
- Optimiser le recouvrement spontané ;
- Améliorer le recouvrement forcé.

I.3 Présentation du projet

I.3.1 Contexte du projet

Après une réunion avec le chef du Service Régional des Ressources et du Système d'Information sur la valeur ajoutée que j'apporterai à l'administration pendant mon stage de fin d'étude. On a convenu le manque d'un outil informatique de gestion du personnel capable de subvenir aux besoins accroissant du service.

I.3.2 Objectifs du projet

L'objectif du projet est de concevoir, réaliser et implémenter une solution de gestion du personnel et leurs affectations aux unités structurelles (niveau hiérarchique) taillée sur le besoin de l'administration, avec la possibilité de partager cette information en temps réel avec l'application de gestion de matériel du centre informatique.

I.4 Conclusion

Ce premier chapitre représente un point de départ pour l'élaboration de ce projet, dans la mesure où il présente successivement l'organisme d'accueil, ses missions et sa vision stratégique ainsi que le cadre général du projet.

Chapitre II :

Gestion de Projet

II.1 Introduction

La gestion de projet ou conduite de projet est une démarche visant à structurer, assurer et optimiser le bon déroulement d'un projet. Elle consiste à appliquer les méthodes, techniques, et outils de gestion spécifiques aux différentes étapes du projet; de l'évaluation de l'opportunité jusqu'à l'achèvement du projet.

II.2 Méthodologie

Pour réussir ce projet, il est nécessaire de se baser sur une méthodologie de travail afin de maîtriser le processus de réalisation de l'application, de garantir la qualité et la fiabilité du logiciel. Pour ce faire le choix s'est posé sur la méthode **Agile** : XP (eXtreme Programming)⁽⁸⁾ car il place le client au cœur du processus de développement et permet de se focaliser sur les besoins les plus prioritaires pour être aussi réactif que possible.

Les valeurs de ce modèle sont :

1. Communiquer quotidiennement avec le client.
2. Privilégier la façon la plus simple pour arriver au résultat.
3. Inciter le client à faire des retours d'information afin d'éviter les changements de dernière minute.
4. Avoir du courage pour effectuer certains changements comme essayer une nouvelle technique, recommencer une itération non validée ou revoir l'organisation du projet.
5. Un respect mutuel avec le client.

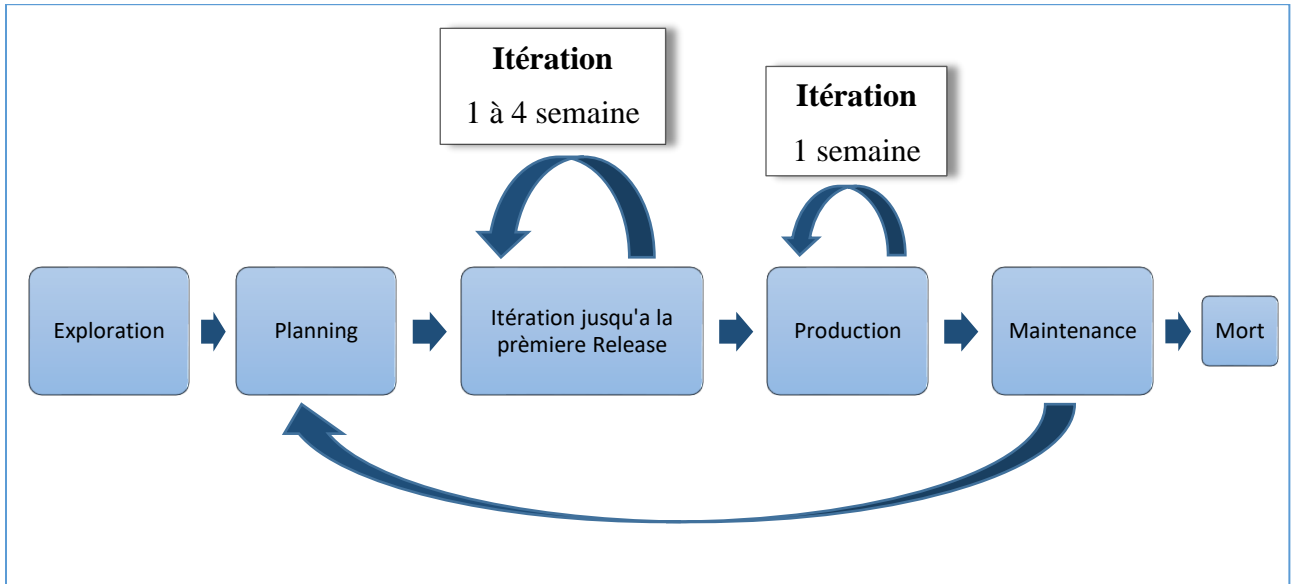
Cycle de vie du schéma XP :

Figure 2 : Schéma de la méthode XP d'Agile

II.3 Approche du projet

Dans cette approche de la méthode XP, Mr le chef du Service Régional des Ressources et du Système d'Information (Maitre d'ouvrage) à donner ces directives au chef de la section des ressources humaines et au chef de la section d'exploitation et système pour m'accompagner tout le long des phases de la réalisation du projet, afin de veiller à ce que je comprenne complètement :

- La méthode de travail du bureau des RH.
- La hiérarchie de la DRI de Marrakech.
- Les listes du personnel à générer.
- Les US qui demandent les données du personnel et les données voulues.
- Les interfaces graphiques web souhaitées.
- Les attestations à imprimer.

Afin d'avoir une solution simple adaptée au besoin réel du service.

II.4 Planification et suivi

L'objectif de la planification du projet est de déterminer les phases du projet ainsi que leurs tâches et durées, ce qui permettra le respect des périodes de validation et délivrer le projet au délai convenu.

Les phases du projet sont comme suit:

- Phase d'étude préalable :

C'est la phase de l'identification des besoins fonctionnels et technique, de la collection des données de démarrage qui devront être chargées sur l'application final. L'identification et justification des choix technologies à utiliser ainsi que la planification des formations sur ces technologies.

- Phase d'analyse et conception :

C'est la phase de modélisation des besoins recensés dans l'étape précédente dans un langage de modélisation, dans notre cas ça sera UML qui est le langage de modélisation graphique de référence dans le domaine du développement logiciel et en conception orientée objet. Dont le but d'avoir une conception graphique, fonctionnelle et technique du projet.

- Phase de réalisation :

Cette phase touchera le côté technique l'application, d'une façon modulaire, en respectant les diagrammes de la phase d'analyse et conception. Le codage des deux parties de la solution (Back-end et Font-end) et enfin le déploiement de l'application dans l'environnement de production.

II.5 Planning

Le planning ci-dessous comprend les étapes du projet et les durées qui leurs ont été allouées aussi bien que les durées des différents sous tâches composant chaque phase. Ce planning a donc servi de tableau de bord pour faire le suivie de l'état d'avancement de la réalisation du projet.

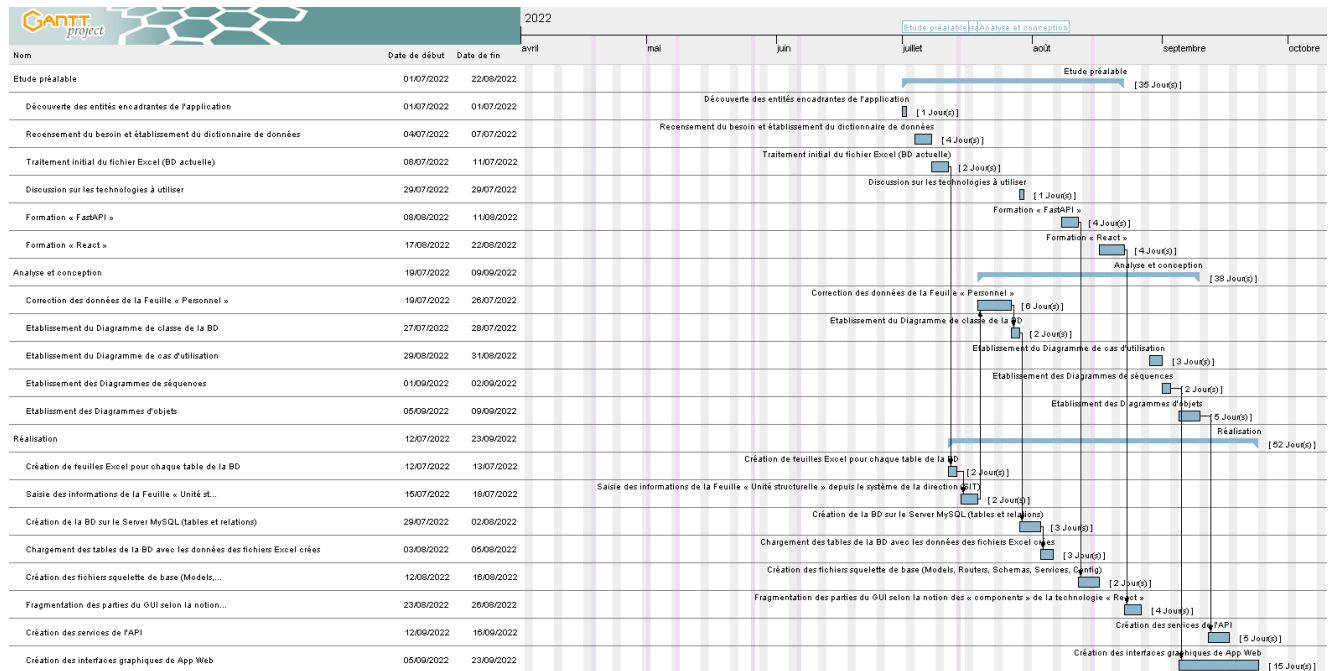


Figure 3 : Diagramme de Gantt du Planning

Cette tâche de planning a été effectuée en utilisant le programme Gantt Project ⁽⁹⁾.

C'est un logiciel libre de gestion de projet écrit en Java, ce qui permet de l'utiliser sur divers systèmes d'exploitation (Windows, Linux, MacOS).



II.6 Conclusion

Ce chapitre a pour objectif de donner une vision sur la démarche de « gestion de projet » par laquelle ce PFE a été réalisé. En passant par la méthodologie de gestion, l'approche du projet jusqu'à la planification et planning du déroulement du projet.

Chapitre III : Etude de l'existant

III.1 Introduction

Ce chapitre dévoilera la méthode de gestion du personnel utilisée par le Bureaux des Ressources Humaines et la problématique qu'elle engendre, ainsi que l'architecture technique disponible pour se mettre dans le cadre environnemental de l'administration afin de mieux choisir l'approche technique à adopter.

III.2 Méthode de gestion utilisée

Pour la gestion du Personnel est son suivie le Bureaux des Ressources Humaines se basait sur un fichier « Excel » contenant tous les informations sur les fonctionnaires. Ce fichier est partagé entre les gestionnaires du bureau RH, chaque modification nécessitait l'ouverture du fichier en mode écriture, ce qui le verrouiller pour les autres collègues.

Une fois le fichier cumul un nombre important de modifications, il est envoyé au responsable de la « Section Matériel et Logistique » pour faire le suivie des personnes non-dotées de machine de travail ainsi qu'un inventaire de périphérique par personne.

Une fois par mois un fichier est envoyé au Responsable RH du service central de la DGI pour l'informer des changements effectués au niveau régional.

Le fichier contient des classeurs des « états à suivre », ex :

- Personnel à la portée d'une mise à la retraite ;
- Personnel en congé de maternité ;
- Liste des agents communaux ;
- Liste des détachés de l'administration.

En plus de statistiques sur le personnel, ex:

- Le nombre de personnes par bâtiments de l'administration et par service ;

- Le pourcentage de l'intégration féminine par apport à celle masculine ;
- Les unités structurelles vacantes ;
- Les postes de responsabilité vacants.

III.3 Problématique

L'étude de la méthode de gestion utilisée discerne clairement les limites de cette façon de gestion. Avec plus de **500 personnes** dispersées sur **17 bâtiments** dans **11 villes** du périmètre de la DRI de Marrakech, On ne peut se permettre de restreindre le traitement des données à une seule personne à la fois, ou de retarder le partage d'information critique à certaine unité de gestion pour effectuer leurs tâches quotidiennes. Sans parler de la perte de temps pour l'élaboration de statiques pouvant être générées automatiquement.

Il est donc primordial d'instaurer un système de gestion capable de relever les défis précités, répandant à l'ampleur du besoin illustré.

Ce besoin à la fois logique et technique, doit prévoir une manière de partager les données nécessaires avec l'application de « Gestion du Matériel » en utilisation.

III.4 Environnement technique disponible

La DRI de Marrakech dispose d'un local technique dans chacun de ces bâtiments. Les serveurs régionaux sans dans le local du bâtiment sis à « TARGA », relié aux autres bâtiments par un réseau privé interne.

Dans ce local technique qui constitue la DMZ⁽¹⁰⁾ du réseau local de la DRI de Marrakech les serveurs suivants sont disponibles :




- Serveur de « Gestion de Courrier » : serveur IIS⁽¹¹⁾ hébergeant une application de gestion des courriers du bureau d'accueil de la DRI Marrakech programmée en ASP⁽¹²⁾.
- Serveur d'Antivirus : « Kaspersky Security Center » qui fournit la protection antivirale aux machines du parc informatique.

- Serveur de base de données : Hébergeant une base de données MySQL ⁽¹³⁾ de l'application « Gestion de courrier » ainsi qu'un ancien serveur de base de données « Ingres ⁽¹⁴⁾ ».
- Serveur d'impression : Doté d'un system « Windows Serveur 2012 R2 » et du logiciel « Megaform ⁽¹⁵⁾ » pour l'impression des émissions en masse des avis d'imposition.

Seuls les trois premiers serveurs sont visibles aux cites déportés.

III.5 Etude comparative ⁽¹⁶⁾

L'étude de marché effectuée est synthétisé dans le tableau ci-dessous représentant un comparatif des 5 solutions applicatives les mieux notées :

Logiciel	Type licence	Besoin de connexion internet	Paramétrage	Point fort
 Avature	Commerciale	Oui, service en ligne seulement	Non adapté	CRM, pour « Candidate Relation Management », en cours de recrutement ou déjà au sein de l'entreprise.
 Factoriel SIRH	Démonstration de 14 jours, Tarifs selon le pack choisi	Oui, SAAS	Complexe	une solution RH complète qui permet de gérer toutes les facettes des ressources humaines d'une entreprise.
 Monday	Commerciale avec période d'essai	Oui	Faible	L'organisation de la répartition des tâches ainsi que le suivi des objectifs pour améliorer de la productivité.



 UKG READY	Commercial	Oui, 100% en ligne	Impossible	Un Tableau de bord statistique complet qui permet de réaliser un suivi des compétences, et de réaliser un suivi des formations en interne avec gestion des participants et résultats.
 Leapsome	Commercial	Ouis, SaaS (en ligne, basé sur le cloud)	Complexe	Permet de réaliser un suivi complet des employés avec la possibilité d'intégration des logiciel de productivité comme Slack, Jira, Teams.

Tableau 1 : Etude comparative des 5 solutions de gestion du personnel du marché 2022

Après l'étude des possibilités trouvées, en conclu que les solutions disponibles sont tous payantes donc des frais supplémentaires (mensuel ou annuel) avec l'obligation d'avoir recours à Internet et aucunement adaptable avec le besoin réel cité dans la partie problématique.

III.6 Conclusion

Ce chapitre à décrit l'état existant de la gestion utilisée par les responsables des ressources humaines ainsi que les besoins et les problèmes rencontrés au quotidien, en plus de l'environnement technique disponible à l'accueil d'une solution applicative. Sans oublié une étude des solutions existantes.

Chapitre IV : Analyse et conception

IV.1 Introduction

La partie « analyse et conception » constitue la phase de départ de toute solution à développer. Le but étant d'exposer une documentation complète sur les différentes architectures de chaque partie. Ce chapitre touche deux principaux volets : le premier touchant l'analyse des besoins des différents acteurs et le deuxième concerne la spécification de ces besoins via des diagrammes UML ⁽¹⁷⁾.

IV.2 Analyse des besoins

IV.2.1 Identification des acteurs

Le tableau ci-dessous illustre les différents acteurs du système :

Acteur	Rôle dans le système
Personnel	Peut avoir le droit de consultation du personnel ou les unités structurelles.
Responsable RH	C'est le principal utilisateur de l'application, l'un des gestionnaires de la section des ressources humaines.
Administrateur	C'est le plus haut profil du système, en plus des autres droits, il gère les profils utilisateurs ainsi que les tables (Pays, Région et Ville)

Système de Gestion Matériel	C'est un acteur externe, c'est le système de la « gestion matériel » qui aura l'autorisation de communiquer avec la partie backoffice de l'application.
------------------------------------	---

Tableau 2 : Les acteur et leurs rôles dans le système

IV.2.2 Spécification des besoins fonctionnels

Les besoins fonctionnels sont les actions et réactions que le système doit faire suite à la demande d'un acteur. L'application doit permettre aux acteurs de :

Personnel :

- Si autorisé, peut consulter les informations du personnel.
- Si autorisé, peut consulter les informations des unités structurelles ainsi que la hiérarchie de la DRI.

Responsable RH :

- Consulter ainsi que traiter les données du personnel (Ajout, modification, suppression).
- Consulter ainsi que traiter les unités structurelles (Ajout, modification, suppression).
- Affecter un personnel à un bâtiment.
- Gérer les types d'unités structurelles
- Gérer les fonctions du personnel.
- Gérer les grades du personnel.
- Gérer les spécialités des diplômes.
- Gérer les niveaux d'étude du personnel.
- Consulter l'historique des anciennes affectations d'un employé, des fonctions, des grades, des postes, des diplômes est niveaux d'étude.
- Consulter un tableau de bord des statistiques RH.
- Imprimer les attestations de travail.

- Pouvoir télécharger au format PDF, l'arborescence de la hiérarchie de la DRI.
- Pouvoir télécharger au format Excel l'ensemble de la situation du personnel.
- Etre notifié des mises à la retraite à prendre en considération, des mises en disponibilité qui arrive à expiration de délai.

Administrateur :

- Gérer les droits d'accès.
- Gérer les données des tables (Pays, Région et Ville).

Système de Gestion Matériel :

- Accéder aux données du personnel.
- Accéder la liste des US.

IV.2.3 Spécification des besoins non fonctionnels

Les besoins non fonctionnels décrivent toutes les contraintes techniques, ergonomiques et esthétiques, auxquelles est soumis l'application.

En ce qui concerne notre solution, nous avons dégagé les besoins suivants :

La simplicité : l'utilisateur doit trouver l'application simple à utiliser.

La rapidité : le système doit répondre dans un temps optimal.

La fiabilité : les données fournies par l'application doivent être fiables.

L'ergonomie : offrir des interfaces conviviales et intuitives.

L'extensibilité : l'habilitation d'ajout de nouveaux modules et fonctions.

La sécurité : seuls les personnes autorisées doivent accéder au système.

L'intégrité : garantir la conformité et l'intégrité des données.

La disponibilité : l'application doit être disponible pour être utilisée par n'importe quel utilisateur.

IV.3 Conception

Pour la représentation des besoins identifiés précédemment nous avons adopté le langage UML⁽¹⁷⁾.



Unified Modeling Language ou Langage de Modélisation

Unifié est : un langage de modélisation graphique à base de pictogrammes conçu comme une méthode normalisée de visualisation dans les domaines du développement logiciel et en conception orientée objet. Il est destiné à faciliter la conception des documents nécessaires au développement d'un logiciel orienté objet, comme standard de modélisation de l'architecture logicielle. Les différents éléments représentables sont :

- Activité d'un objet/logiciel
- Acteurs
- Processus
- Schéma de base de données
- Composants logiciels
- Réutilisation de composants.

Afin de dessiner les diagrammes du modèle UML et aussi de pouvoir générer le code initial de la base de données automatiquement, nous avons utilisé le logiciel PowerDesigner⁽¹⁸⁾.



Anciennement appelé (PowerAMC), il est un logiciel de conception créé par la société SAP⁽¹⁹⁾, qui permet de modéliser les traitements informatiques et leurs bases de données associées. Par défaut, PowerDesigner stocke ses modèles sous forme de fichiers, dont l'extension dépend du type de modèle:

BPM ou **CDM**. La structure interne du fichier peut être du XML ou du binaire compressé.

IV.3.1 Diagrammes de cas d'utilisation

La figure ci-dessous représente le diagramme de cas d'utilisation de l'acteur « Responsable RH » qui sera présenté comme package « UseCaseDiagram_RespRH » dans le diagramme de cas d'utilisation global:

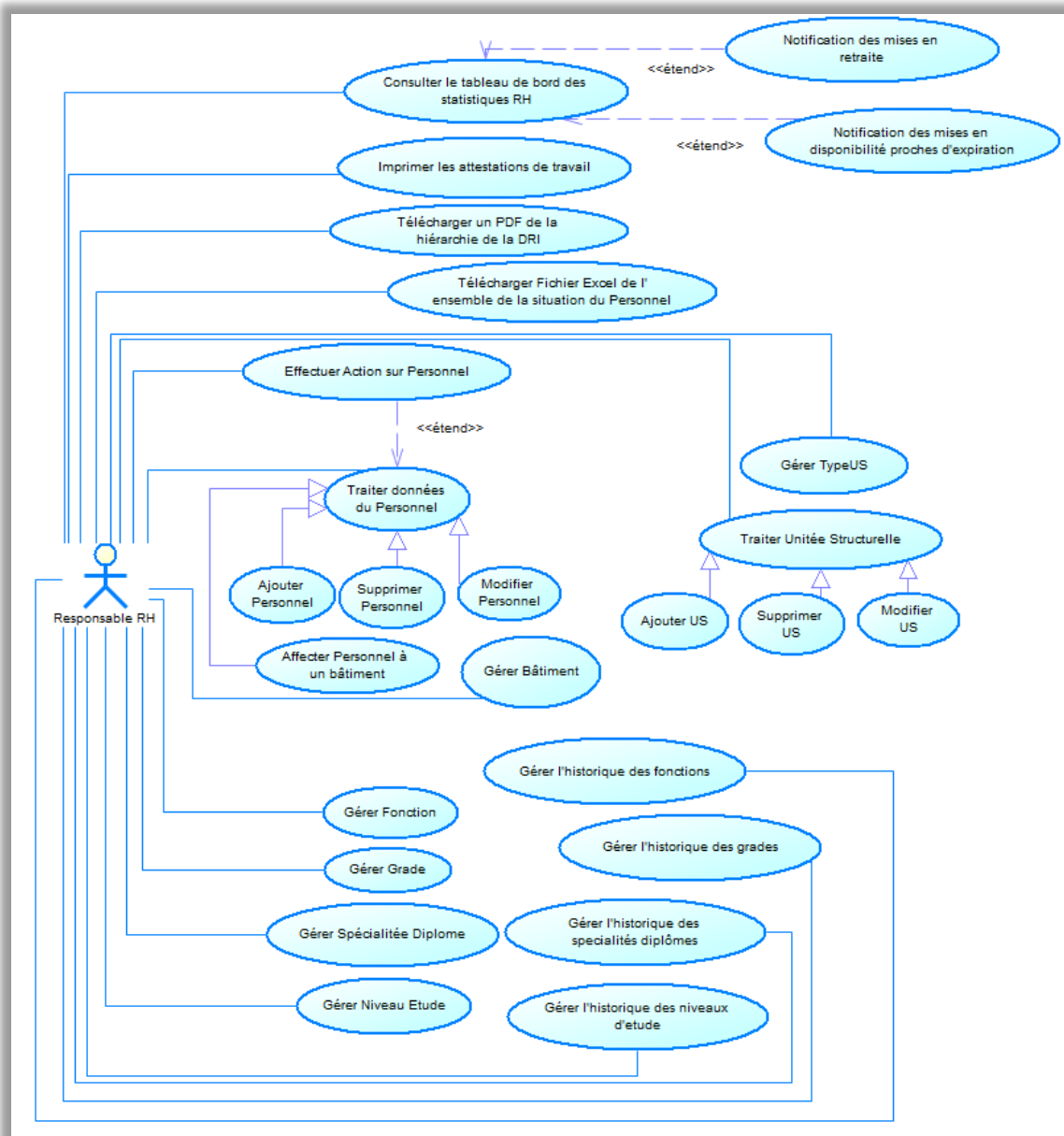


Figure 4 : Diagramme de cas d'utilisation de l'acteur "Responsable RH"

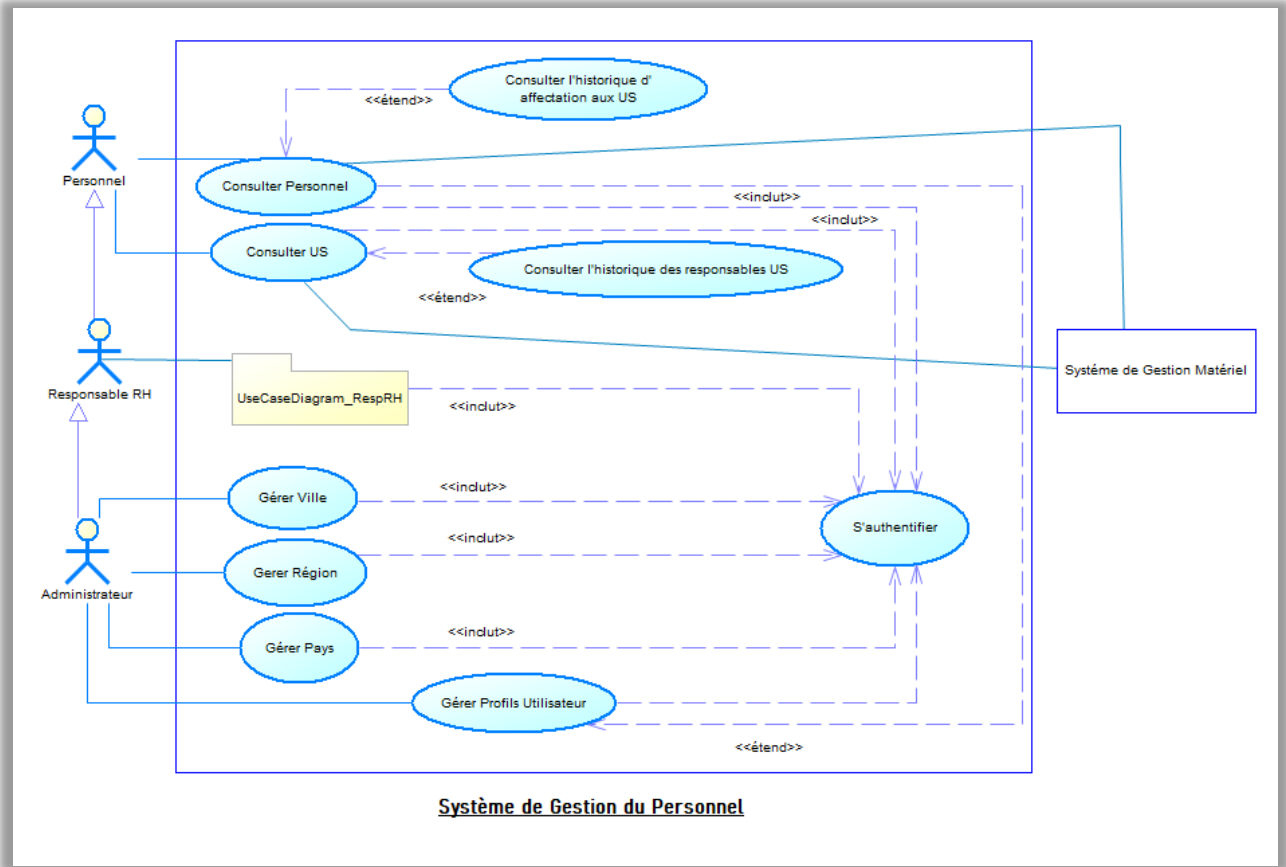


Figure 5 : Diagramme de cas d'utilisation global de l'application

IV.3.2 Diagrammes de séquences

Les diagrammes de séquences permettent de montrer les interactions d'objets dans le cadre d'un scénario d'un diagramme des cas d'utilisation. Dans un souci de simplification, on représente l'acteur principal à gauche du diagramme, et les acteurs secondaires éventuels à droite du système. Le but est de décrire comment se déroulent les interactions entre les acteurs ou objets.

Ci-dessous les diagrammes de séquences clés :

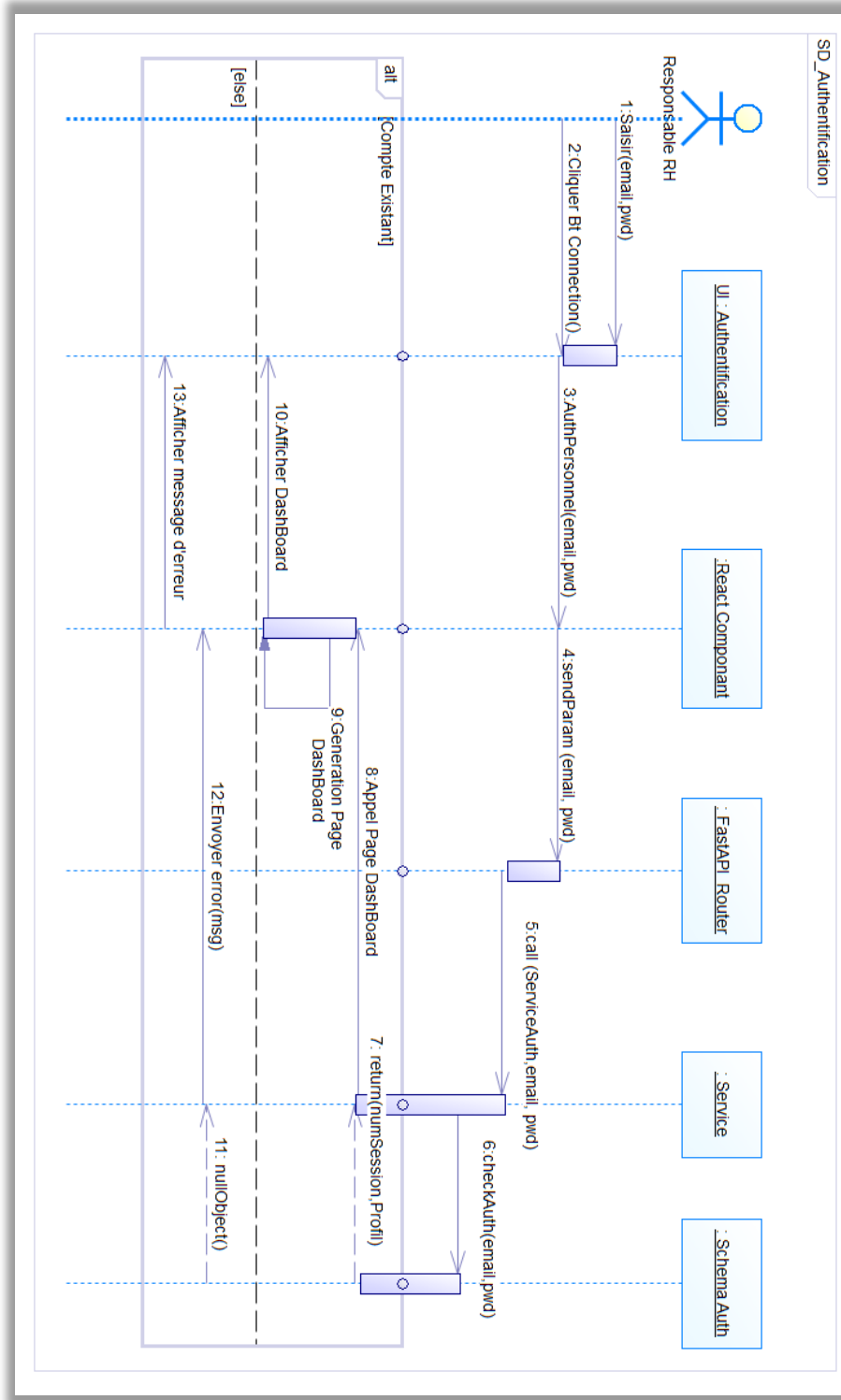


Figure 6 : Diagramme de séquence du cas d'utilisation "Authentication"

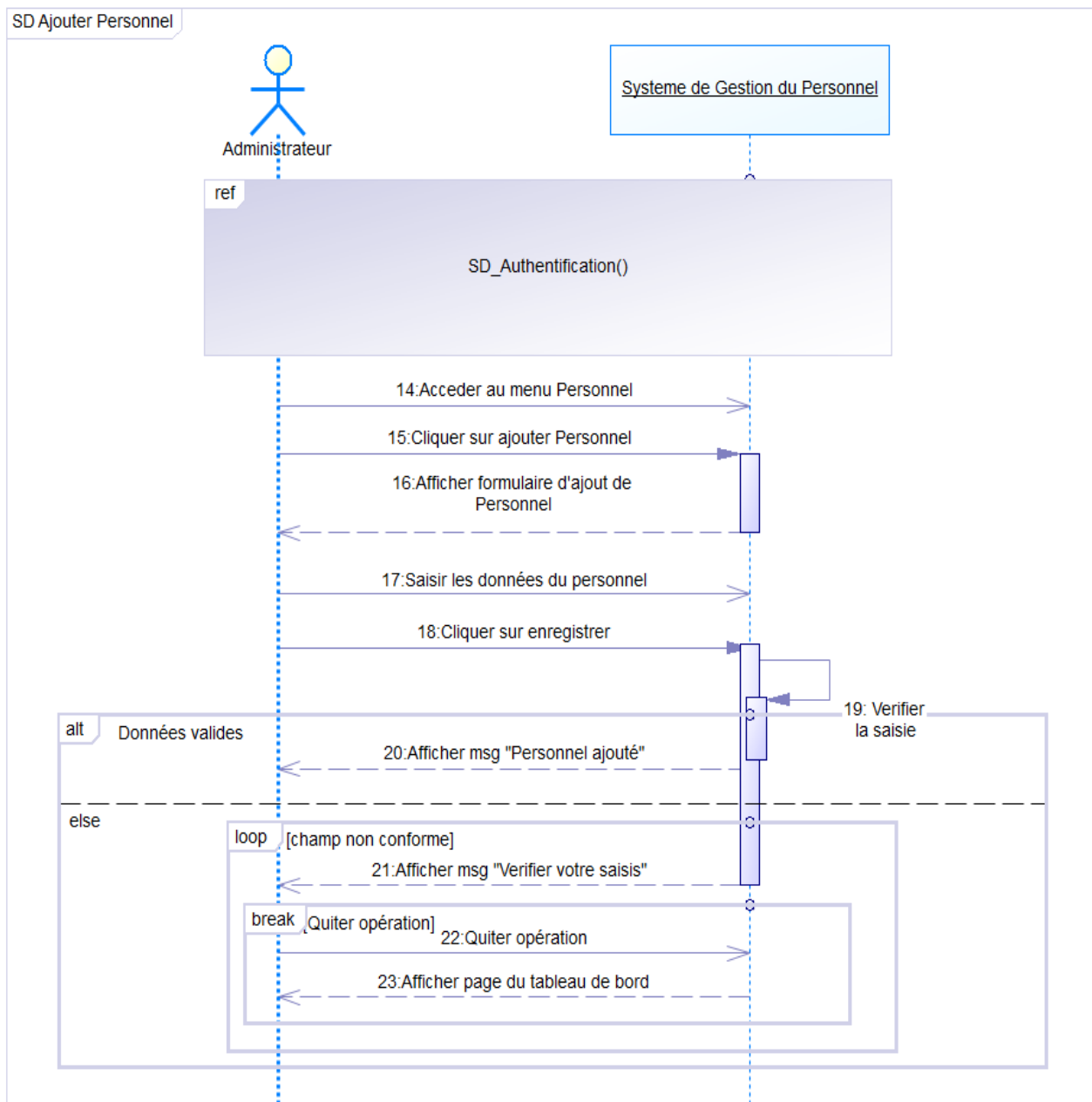


Figure 7 : Diagramme de séquence du cas d'utilisation "Ajouter Personnel"

IV.3.3 Diagramme de classes

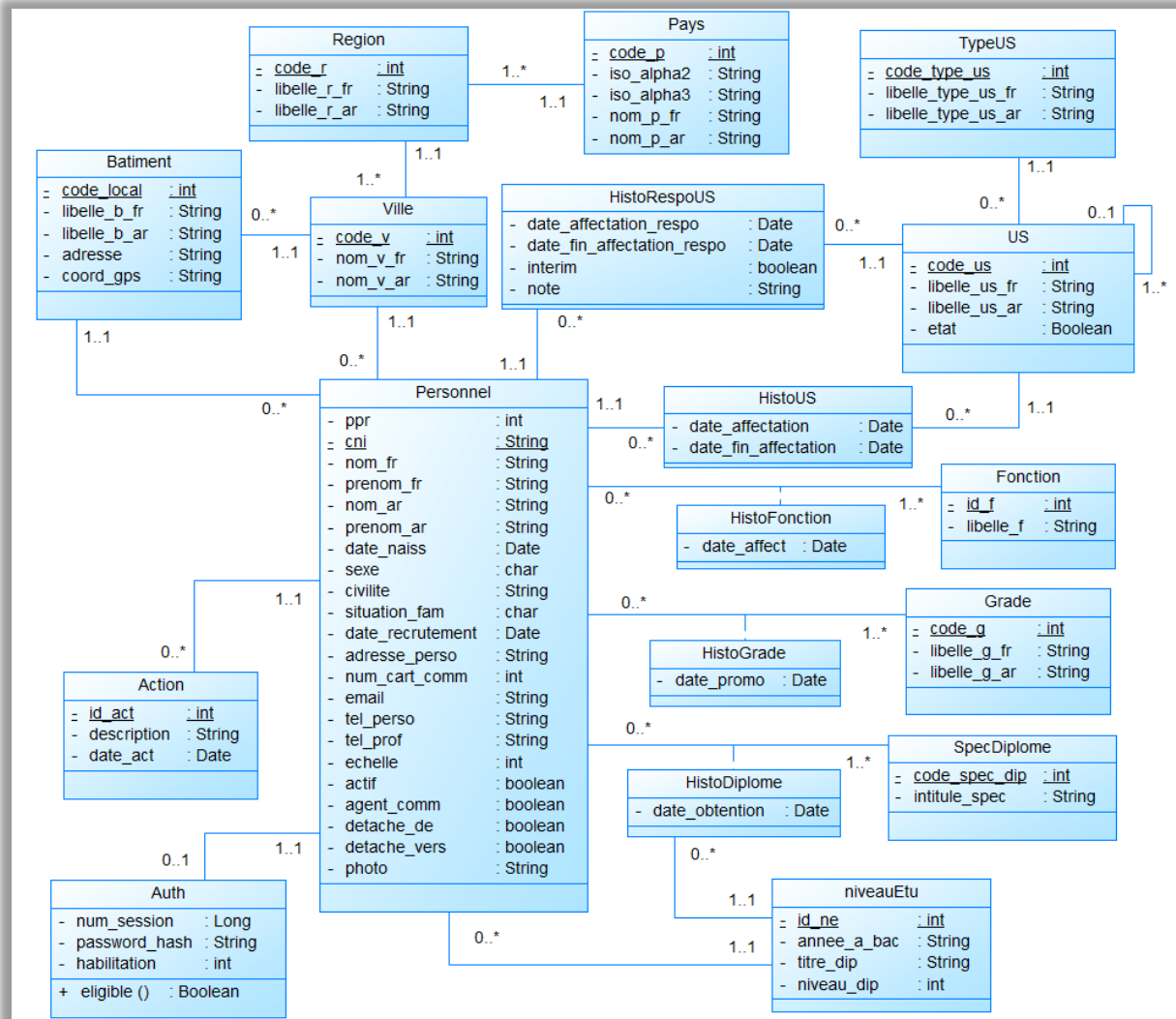


Figure 8 : Diagramme de classes de l'application

A partir du diagramme de classes ci-dessus, nous avons généré le schéma de la base de donnée ci-dessous, qui sera transformé en code source SQL prêt à être exécuté sur le server MySQL

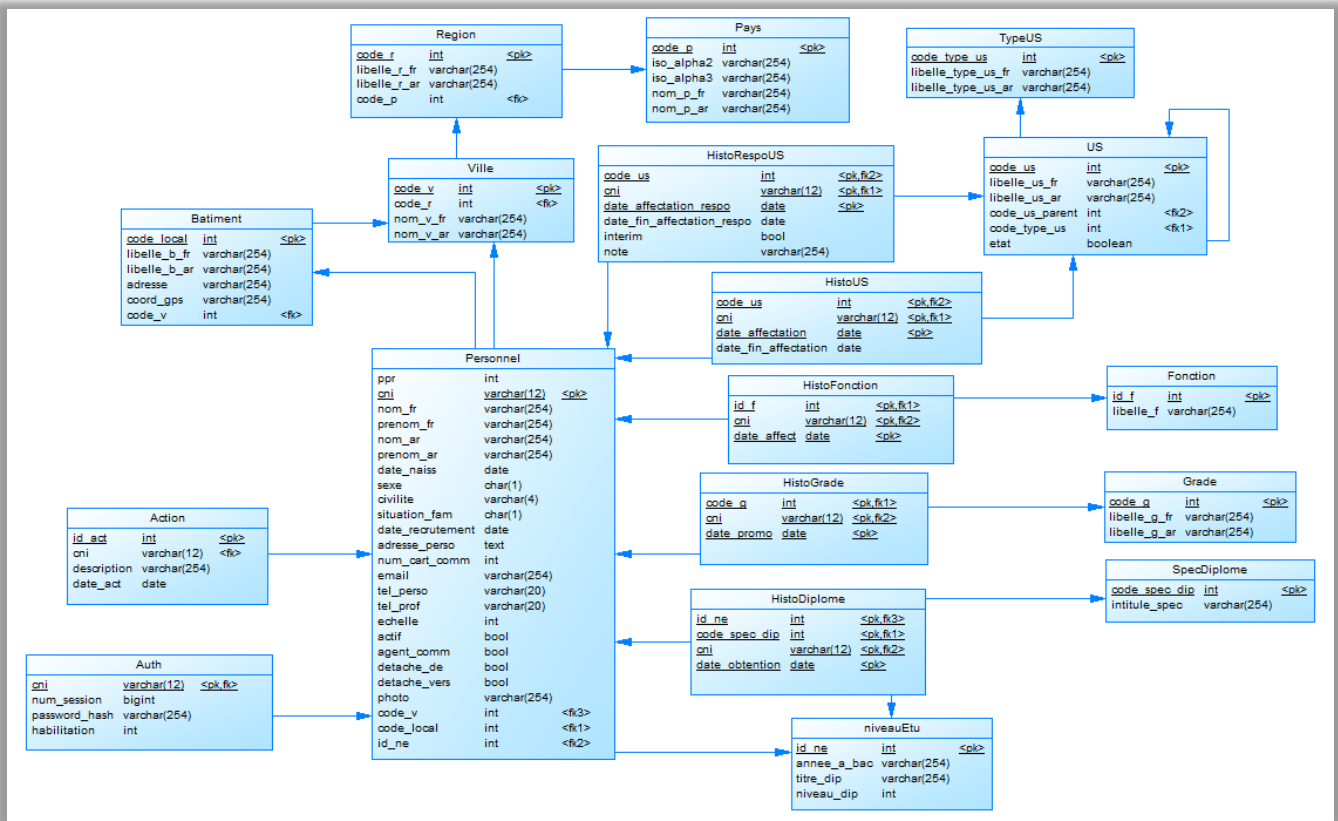


Figure 9 : Schéma de la base de données

IV.4 Conclusion

Dans ce chapitre nous avons procédé à l'identification des acteurs ainsi que la spécification des besoins fonctionnels et non fonctionnels de l'application. Et la réalisation des diagrammes UML utiles à notre conception.

Chapitre V :

Réalisation

V.1 Introduction

Ce chapitre constitue le dernier volet de ce rapport ayant pour objectif d'exposer le métier du travail réalisé. Pour ce faire, nous allons présenter en premier temps les outils de développement ainsi que le choix des technologies et l'architecture utilisées. Et en deuxième temps des exemples de code avec quelques captures d'écrans des interfaces graphiques clés de la solution.

V.2 Architecture, technologies et outils de développement

V.2.1 Architecture

En informatique, le terme « architecture » désigne la structure générale inhérente à un système informatique, l'organisation des différents éléments du système et les relations entre ces éléments. Cette structure fait suite à un ensemble de décisions stratégiques prises durant la conception de tout ou partie du système informatique.

Dans notre cas, nous avons adopté une architecture à trois niveau pour répartir la logique applicative sur plusieurs machines dans le but d'exploiter pleinement les ressources matériels et logiciels disponibles, et d'avoir des services à pleine capacité.

Parmi les avantages de l'architecture à plusieurs niveaux:

- La facilité de maintenance : isole la maintenance et empêche les effets en cascade;
- L'évolutivité : Mettez à l'échelle des niveaux séparés sans toucher à d'autres niveaux
- La flexibilité : S'agrandit de n'importe quelle manière selon les besoins
- La sécurité : Chaque niveau peut être sécurisé séparément et de différentes manières

Détail des niveaux de l'architecture utilisée :

- Niveau Présentation : est l'interface utilisateur et la couche de communication de l'application, où l'utilisateur final interagit avec l'application. Sa principale fonction est d'afficher des informations à l'attention de l'utilisateur et d'en collecter de ce dernier, c'est la partie Font-End de notre solution (L'application Web).
- Niveau Application : également appelé niveau logique ou niveau intermédiaire, est le cœur de l'application. Dans ce niveau, les informations collectées dans le niveau Présentation sont traitées, parfois par rapport à d'autres informations, dans le niveau Données, en utilisant la logique applicative qui est l'ensemble des règles métier. Le niveau Application peut également ajouter, supprimer ou modifier des données dans le niveau Données, c'est la partie Back-End de notre solution (L'API).
- Niveau Données : parfois appelé niveau base de données ou niveau accès aux données, est l'endroit où les informations traitées par l'application sont stockées et gérées.

V.2.2 Technologies

Pour chacun des niveaux de l'architecture présentés dans la rubrique ci-dessus, le choix des serveurs, langages de programmation et Framework ⁽²⁰⁾ est synthétisé dans le tableau ci-dessous :

Niveau	Serveur	Langage de Programmation	Framework
Présentation	Node.js	Javascript ⁽²¹⁾	React ⁽²²⁾
Application	Uvicorn	Python ⁽²³⁾	FastAPI
Données	MySql ⁽¹³⁾	SQL	-

Tableau 3 : Langages de programmation, Serveurs et Framework utilisés

Choix des serveurs :

Chaque serveur est hébergé dans une machine à part. Le choix du serveur « MySQL » pour la base de données a été pris car l'administration est prédisposée d'une machine dotée de ce dernier. Le serveur « uvicorn » est installé sur une machine Windows, car on a opté pour l'utilisation du langage « Python » et le serveur « node.js » est installé sur le post du responsable du bureau RH pour exécuter le code des interfaces graphiques dont bénéficiera son équipe.

Choix des langages de codage :

Puisque l'interface homme-machine est une application web, on ne peut contourner l'utilisation de :



: « **HyperText Markup Language** ⁽²⁴⁾ » qu'on peut traduire par « langage de balises pour l'hypertexte ». Il est utilisé afin de créer et de représenter le contenu d'une page web et sa structure.

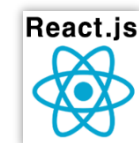


: « **Cascading Style Sheets** ⁽²⁵⁾ » ou feuilles de style en cascade, est un langage informatique qui décrit la présentation des documents HTML et XML, pour rendre les documents HTML intéressants visuellement et captivant en lecture.



: ou **JavaScript** ⁽²¹⁾ est un langage de programmation qui permet de créer du contenu mis à jour de façon dynamique, comme contrôler le contenu multimédia ou animer des images par exemple.

Le développement d'interface web avec les langages ci-dessus prend un temps important. Donc pour gagner du temps, on a utilisé le Template « adminLTE ⁽²⁶⁾ » qui est hautement personnalisable et facile à utiliser en le fragmentant avec :



: aussi appelé **React** ⁽²²⁾ qui est une bibliothèque JavaScript libre, dans le but principal est de faciliter la création d'application web monopage, via la création de composants dépendant d'un état et générant une page (ou portion) HTML à chaque changement d'état.

Pour la partie Back-end nous avons besoin d'un langage dans la syntaxe est simple, compacte, multiplateformes et avec de nombreuses bibliothèques qui seront utiles pour ajouter toutes les fonctionnalités présentes et future.



Python⁽²³⁾ correspond exactement au besoins formulés. Il est un langage de multiplateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet. Il est doté d'un typage dynamique fort, d'une gestion automatique de la mémoire par ramasse-miettes et d'un système de gestion d'exceptions. Il peut s'utiliser dans de nombreux contextes et s'adapter à tout type d'utilisation grâce à des bibliothèques spécialisées, surtout pour l'automatisation des tâches simples mais fastidieuses.

Puisque la partie API est en fin de compte un service web, nous avons choisi le Framework Web le plus apprécié selon l'enquête auprès des développeurs de « Stack Overflow » en 2021.



FastAPI⁽²⁷⁾: est un Framework Web pour développer des API RESTful⁽²⁸⁾ en Python. FastAPI est basé sur Pydantic⁽²⁹⁾ et des indications de type pour valider, sérialiser⁽³⁰⁾ et désérialiser⁽³¹⁾ les données, et générer automatiquement des documents OpenAPI.

Est évidemment le langage SQL pour exploiter la base de données.



: **Structured Query Language**, ou en français langage de requête structurée, qui est un langage informatique normalisé servant à exploiter des bases de données relationnelles. La partie langage de manipulation des données de SQL permet de rechercher, d'ajouter, de modifier ou de supprimer des données dans les bases de données relationnelles.

V.2.3 Outils de développement

Avec cette multitude de langages de codage informatique, l'utilisation d'un éditeur de code puissant et supportant les différentes syntaxes et auto-complétions est un outil primordial à avoir pour réduire la durée du tapage du code.



Visual Studio Code ⁽³²⁾ ou VS Code : est un éditeur de code extensible développé par Microsoft, il est rapide et supporte plusieurs langages et fonctionnalités incluent la prise en charge du débogage, la mise en évidence de la syntaxe, la complétion intelligente du code. Facilement paramétrable pour avoir un environnement de développement personnalisé.

En plus d'un bon éditeur de code, garder des versions des fichiers de code tout le long du développement, surtout dans le cas de modification critique ou la nécessité de revenir en arrière vers une version stable est souhaité. Nous avons besoin d'un logiciel de gestion de versions avec possibilité de sauvegarde en ligne.



⁽³³⁾: Est le logiciel de gestion de versions le plus populaire dans le développement logiciel et web, qui est utilisé par des dizaines de millions de personnes, sur tous les environnements (Windows, Mac, Linux). Offrant la possibilité de synchroniser les répertoires de projets facilement avec des hébergeurs de code informatique en ligne.

V.3 API (Back-end)

V.3.1 Arborescence

<pre> root .env .gitignore dependencies.py main.py README.md requirements.txt +---config database.py settings.py +---models +---routers +---schemas +---services </pre>	<p><u>Contient :</u></p> <ul style="list-style-type: none"> - Les variables d'environnement - les fichiers à ne pas synchronisés avec le serveur git - les fonctions de relai, ex: <code>get_db()</code> pour l'accès à la BD - L'instanciation de l'application back-end - Commentaires d'utilisation - Les bibliothèques à installer <ul style="list-style-type: none"> - Le fonction d'ouverture de la session avec la BD - Les informations de connexion à la BD <ul style="list-style-type: none"> - Les fichiers des classes schémas des tables de la BD <ul style="list-style-type: none"> - Les fichiers des fonctions des liens de l'application <ul style="list-style-type: none"> - Les fichiers des classes des objets de l'application <ul style="list-style-type: none"> - Les fichiers fonctions des requêtes avec la BD
---	---

Tableau 4 : Arborescence de la partie "Back-end"

V.3.2 Captures d'écrans

L'API qui est la partie « Back-end » sur laquelle est basé notre solution, est doté d'une interface graphique accessible en ajoutant [/docs](#) ou [/redoc](#) après l'url d'accès de ce dernier.

Ci-dessous une capture d'écran montrant les méthodes dans dispose le « service personnel » qui a pour objectif de donner « les points de terminaisons » pour l'insertion, la modification, la suppression ainsi que le retour des différents résultats de sélections voulus.

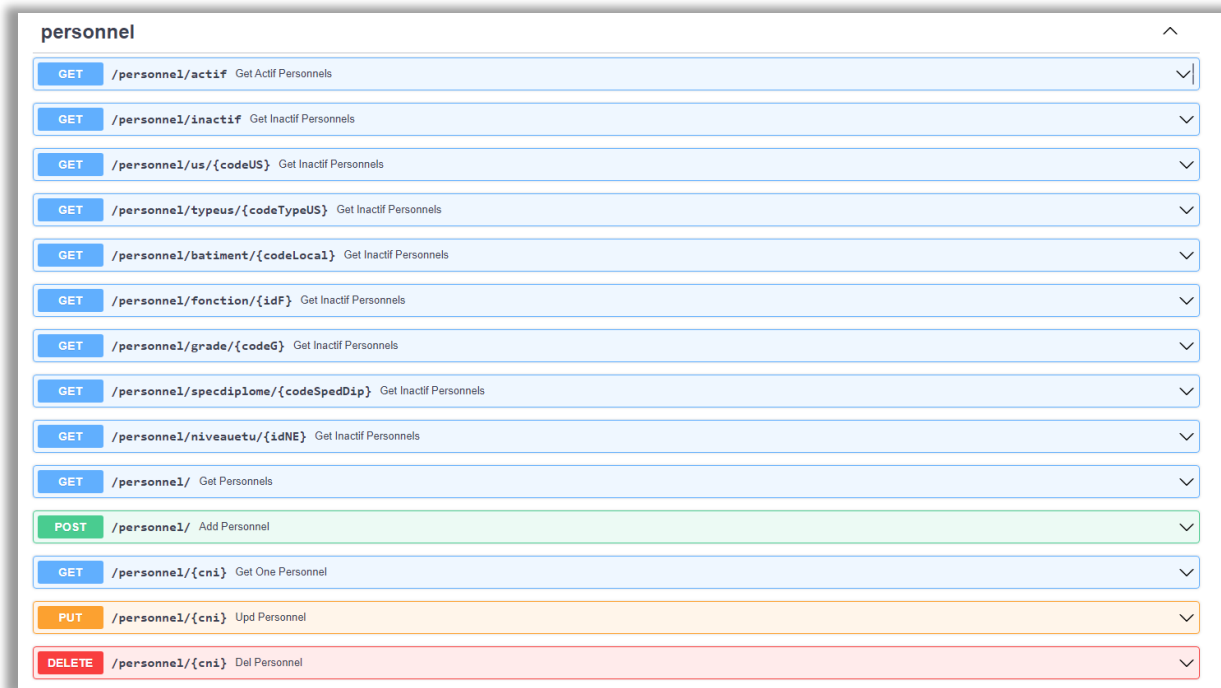


Figure 10 : Capture d'écran des points de terminaison du Service "Personnel" de L'API

L'utilisateur peut cliquer sur n'importe quelle « méthode » pour tester son fonctionnement, en ouvrant la méthode et cliquant sur le bouton « Try it out ».

L'exemple de la capture d'écran ci-dessous test le cas de la sélection du personnel actifs :

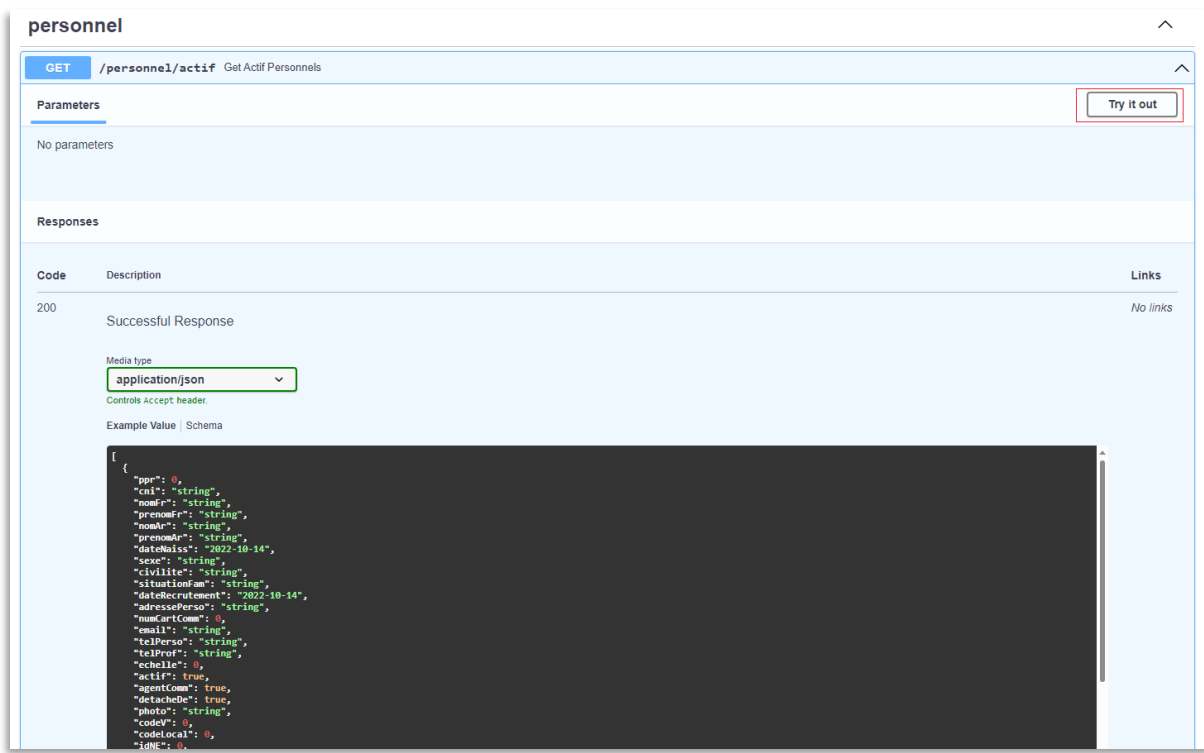


Figure 11 : Capture d'écran du point de terminaison - Afficher tous le personnel actifs

Après le clique sur « Try it out » l'utilisateur à le résultat de la sélection sous forma « JSON » et le lien à utiliser sur les systèmes tiers.

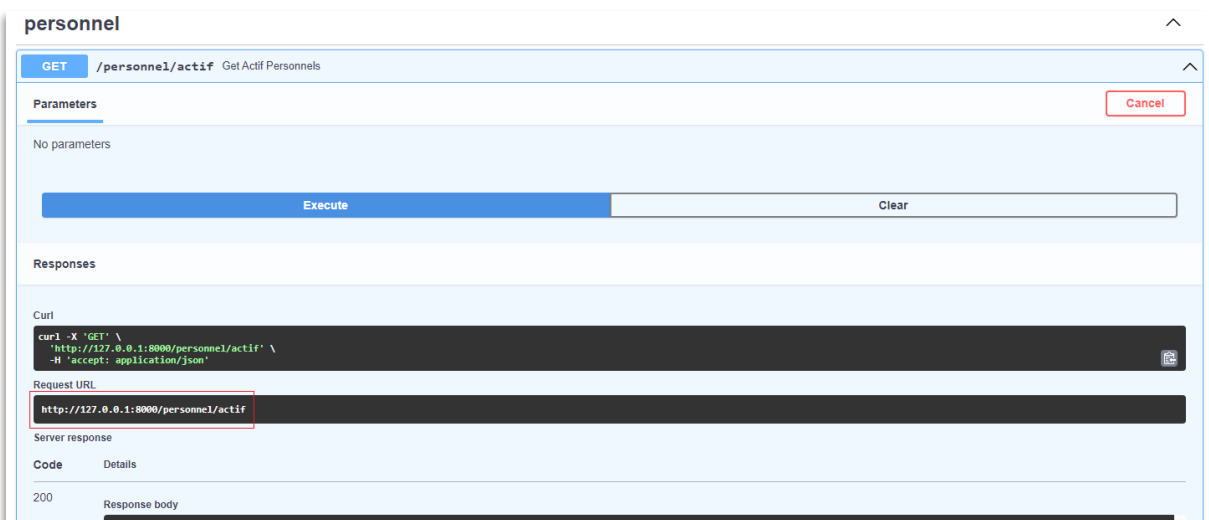


Figure 12 : Capture d'écran du résultat de "Get Actif Personnels"

V.3.3 Exemples de code

Après le démarrage du serveur depuis le répertoire racine de la partie « Back-end » :

```
uvicorn main:app -- start
```

```
from fastapi import FastAPI
from fastapi.middleware.cors import CORSMiddleware

from routers.personnel import router as personnelRouter
# Ainsi que l'importation de tous les autres routers (us, typeus, etc.)

app = FastAPI()

origins = [
    "http://localhost",
    "http://localhost:3000",
]

app.add_middleware(
    CORSMiddleware,
    allow_origins=origins,
    allow_credentials=True,
    allow_methods=["*"],
    allow_headers=["*"],
)

app.include_router(personnelRouter)
# Ainsi que l'inclusion de tous les autres routers (us, typeus, etc.)

@app.get("/")
def root():
    return {"message": "Hello to RH Application"}
```

Figure 13 : Code source - main.py

Le serveur instancie l'application et précise les adresses IP qui peuvent la susciter et les routes disponibles (exemple : /personnel , /us, /typeus, etc.) . Pour chacune de ces routes, il existe un fichier dans le répertoire '/routers' pour spécifier les fonctions à appeler pour avoir le résultat souhaité selon le type de requête HTTP formulée :

GET : pour sélectionner une ressource

POST : pour insérer une ressource.

PUT : pour mettre à jour une ressource.

DELETE : pour supprimer une ressource.

La figure ci-dessous montre un exemple du fichier des routes de 'personnel.py' :

```
from typing import List
from fastapi import APIRouter, Depends, HTTPException
from schemas.personnel import PersonnelCreate, Personnel
from sqlalchemy.orm import Session

from dependencies import get_db
from services.personnelService import
GetAllPersonnel, CreatePersonnel, UpdatePersonnel, DeletePersonnel

router = APIRouter(prefix="/personnel", tags = ["personnel"])

@router.get("/", response_model=List[Personnel], status_code=200)
async def get_personnels(db:Session = Depends(get_db)):
    all_personnel = await GetAllPersonnel(db=db)
    return all_personnel

@router.post("/", response_model=Personnel, status_code=201)
async def add_personnel(personnel : PersonnelCreate, db:Session = Depends(get_db)):
    new_personnel = await CreatePersonnel(db=db, personnel=personnel)
    if new_personnel:
        return new_personnel
    raise HTTPException(status_code=400, detail="Erreur de création !!!")

@router.put("/{cni}", response_model=Personnel, status_code=200)
async def upd_personnel(cni:str, personnel : PersonnelCreate, db:Session =
Depends(get_db)):
    personnel_to_upd = await UpdatePersonnel(db=db, personnel=personnel, cni = cni)
    if personnel_to_upd:
        return personnel_to_upd
    raise HTTPException(status_code=400, detail="Erreur de mise à jour !!!")

@router.delete("/{cni}", status_code=200)
async def del_personnel(cni:str, db:Session = Depends(get_db)):
    result = await DeletePersonnel(db=db, cni = cni)
    if result:
        return {"msg": "Supprimer"}
    raise HTTPException(status_code=404, detail=f"La cni : {cni} est inexistante !!!")
```

Figure 14 : Code source - /routers/personnel.py

Dans le code ci-dessus, on fait appel aux méthodes du service 'personnelService' relatives au traitement du personnel depuis '/services' qui contient tous les fichiers des services disponibles. Le fichier 'personnelService.py' fait appel à la classe quiinstanciera l'objet 'Personnel' depuis le répertoire '/schemas' et au modèle de la table 'personnel' depuis le répertoire '/models', pour l'exécution des opérations CRUD d'une façon prédéfinie.

Pour les requêtes avancées (comme ceux des statistiques du personnel), l'utilisation de requête SQL brute ne peut être contournée, ci-dessous deux requêtes exemple :

```
Select g.codeG,g.libelleGFr,count(*) as NbrPersonnelByGrade
from Personnel p, histograde hg, grade g
where p.cni=hg.cni and hg.codeG=g.codeG and p.actif=1
and hg.datePromo =
(
select hg2.datePromo from histograde hg2 where hg2.cni=p.cni ORDER BY
hg2.datePromo DESC LIMIT 1
)
GROUP BY g.libelleGFr ORDER BY g.codeG ASC
```

Figure 15 : Requête SQL 'Nombre de personnel par grade'

```
select SUM(nbrPCTR) AS nbrPCTR
from
(
SELECT COUNT(*) as nbrPCTR
FROM `personnel` p
WHERE p.actif=1
GROUP BY dateNaiss
HAVING date_add(p.dateNaiss, interval 62 YEAR) <= DATE_ADD(CURDATE() ,
INTERVAL 6 MONTH)
)
AS nbrPCTR
```

Figure 16 : Requête SQL 'Nombre de personnel prêt de la retraite'

V.4 Application Web (Front-End)

V.4.1 Arborescence

<pre> root .env .gitignore package-lock.json package.json README.md +---node_modules +---public favicon.ico index.html manifest.json robots.txt +---pictures \---template \---src App.js index.js +---components MainFooter.js MainSidebar.js Navbar.js \---tools loadPages.js Pagination.js \---pages </pre>	<p>Contient :</p> <ul style="list-style-type: none"> - Les variables d'environnement - les fichiers à ne pas synchronisés avec le serveur git - Générer automatiquement par NPM à but de déploiement - Générer automatiquement par NPM - Commentaires d'utilisation - Les fichiers des classes de REACT - Les ressources publics de l'application - l'icône des pages de l'application - La page HTML de l'application - les informations sur l'application (nom auteur, icône, etc.) - Les pages à ne pas être indexées par les moteurs de recherche - Les photos du personnel de l'application - Tous les ressources de style (Feuilles CSS ...etc.) - Les fichiers corps de l'application - L'instanciation de l'application front-end - Le code de création du visuel de la page HTML « index » - Répertoire des composants d'une page - Fragment du pied de page - Fragment du menu latéral - Fragment du menu - Répertoire des composants d'aides - Fichier de la génération de la pagination - Fichier d'affichage de la pagination - Répertoire des pages de l'application web
---	---

Tableau 5 : Arborescence de la partie "Front-end"

V.4.2 Captures d'écrans

La première page qui s'ouvre lors de l'accès à l'application est la page d'accueil, qui va permettre à l'utilisateur de l'application de s'authentifier pour accéder à son profil. Elle permettra aussi aux personnel non-disposants de compte d'en demander un, ou de renouveler le mot de passe pour ceux qui l'ont oublié.

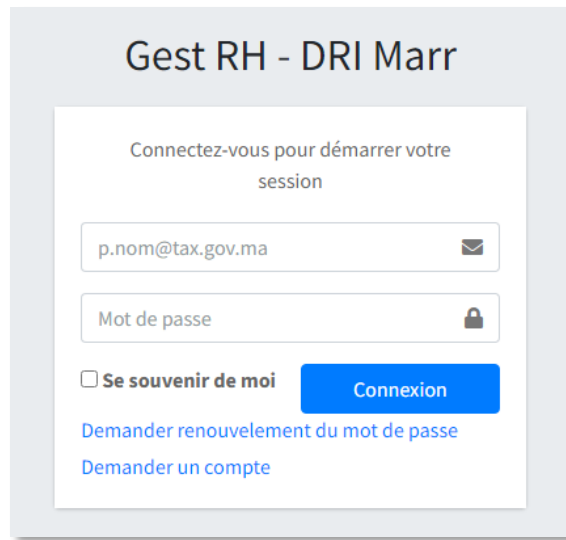


Figure 17 : Capture d'écran de la page d'authentification

Après l'authentification le menu s'affiche selon le profil de l'utilisateur (Administrateur, Responsable RH ou Consultation).

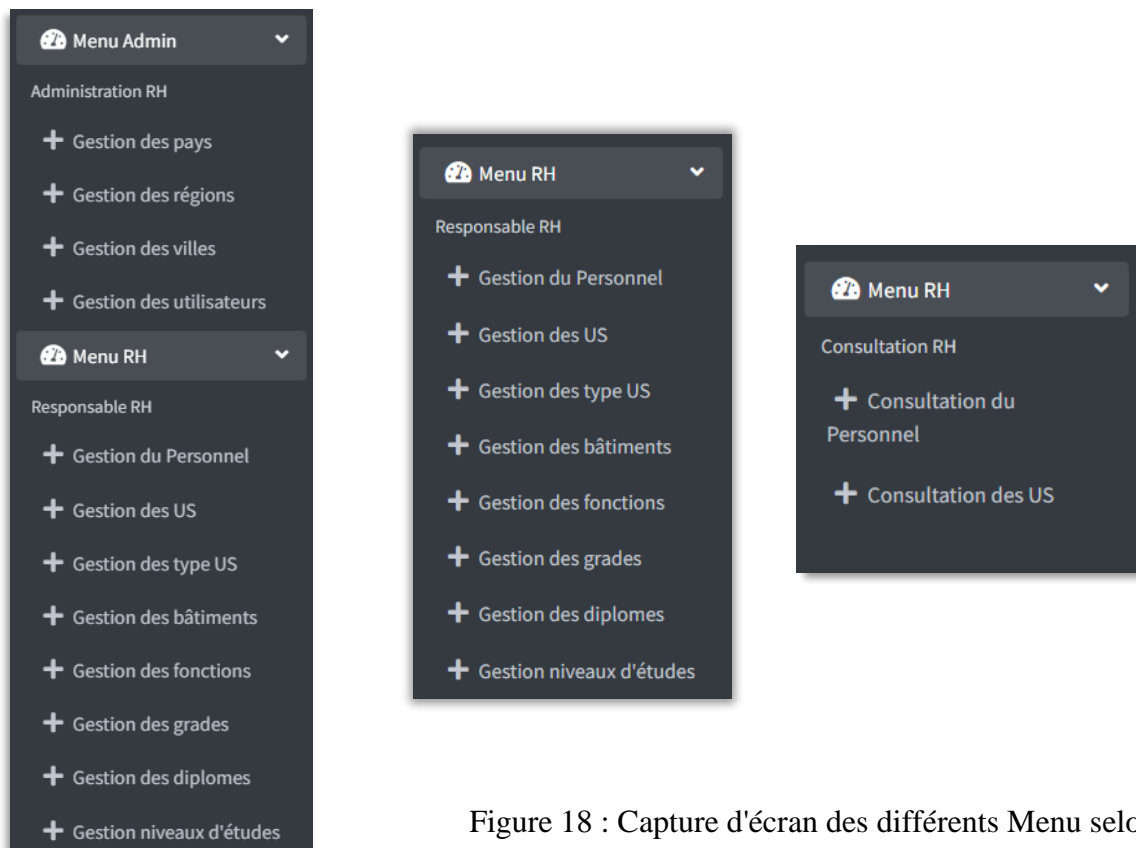


Figure 18 : Capture d'écran des différents Menu selon le profil

La page principale de l'application web et le 'tableau de bord' qui récapitule les chiffres clés de cette gestion sous forme de statistique pour permettre au gestionnaire RH d'avoir une vue globale sur les ressources disponibles.

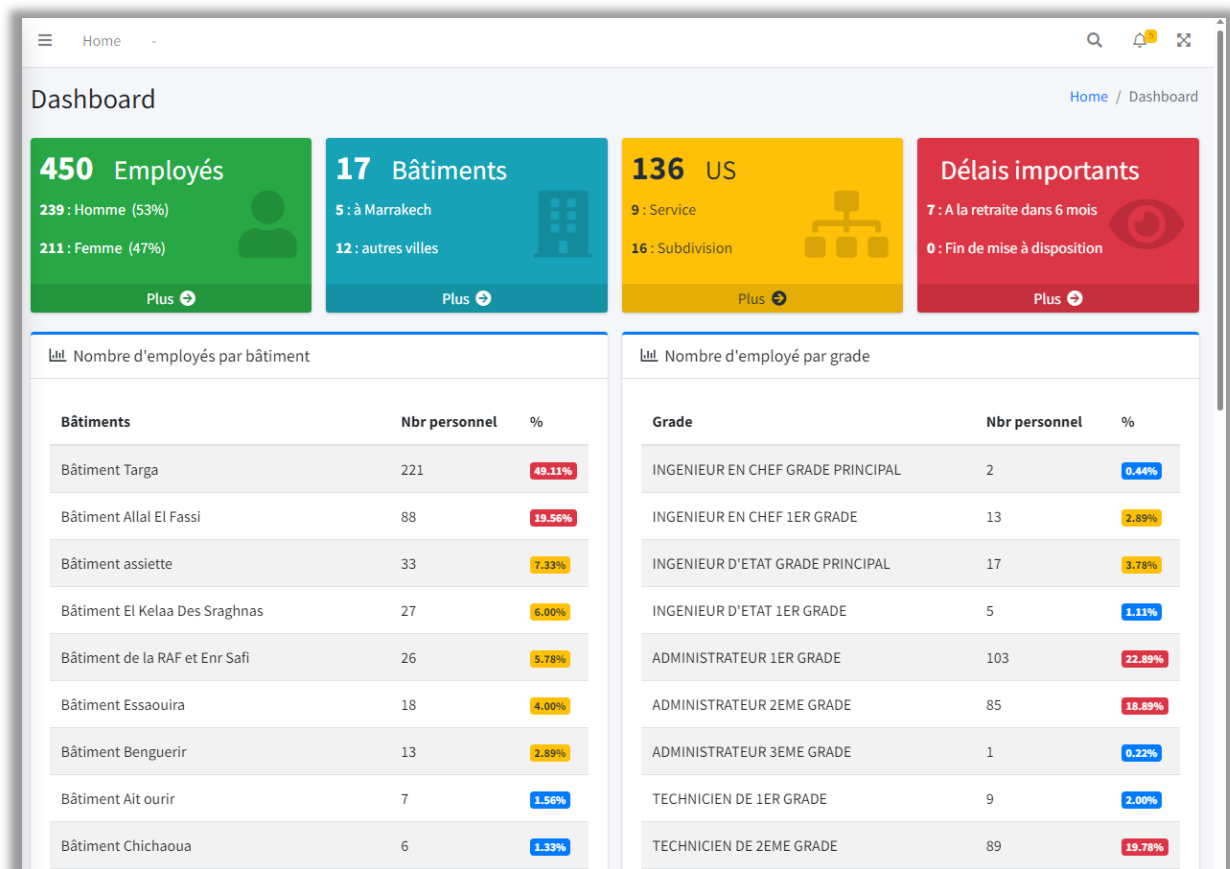


Figure 19 : Capture d'écran de la page du tableau de bord

Lors de la clique sur l'élément 'Gestion des US' du menu, la page de gestion des unités structurelles s'affiche organisant les US sous forme d'arborescence indentée pour faciliter la structuration de ces dernières. Cette page permet aussi de télécharger la liste de ces US sous format PDF, Excel ou encore de les imprimer.

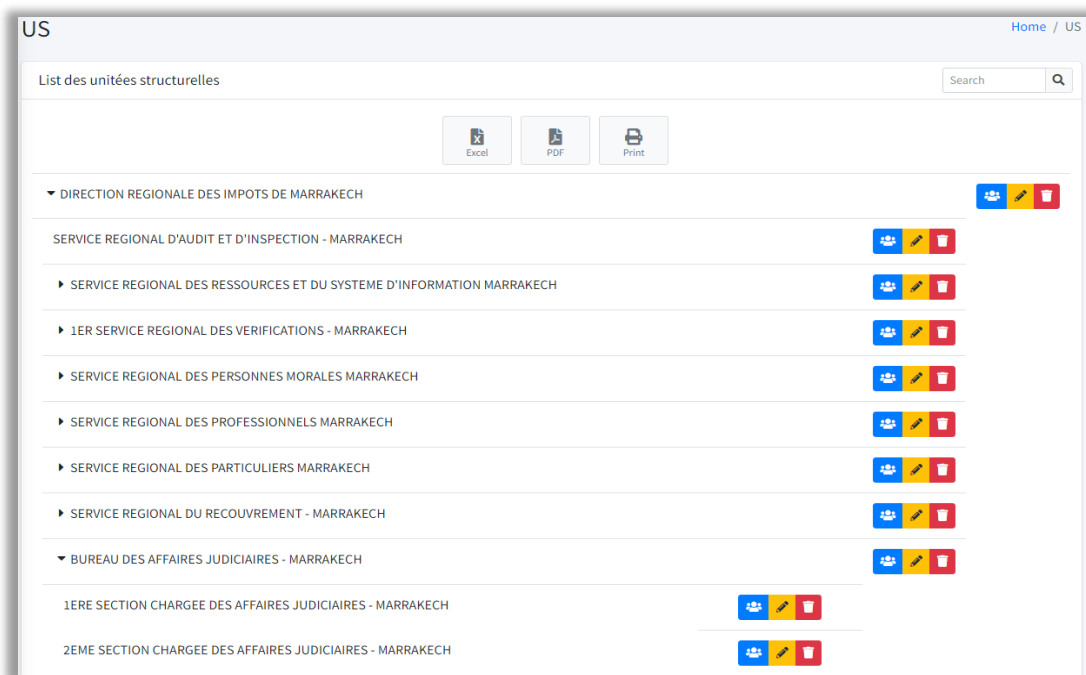


Figure 20 : Capture d'écran de la page 'Gestion des US'

La page 'Gestion du Personnel' affiche les identifiants du personnel ainsi que leurs nom et prénom sous forme d'un tableau avec la possibilité de navigation, d'export et de traitement sur le personnel sélectionné.

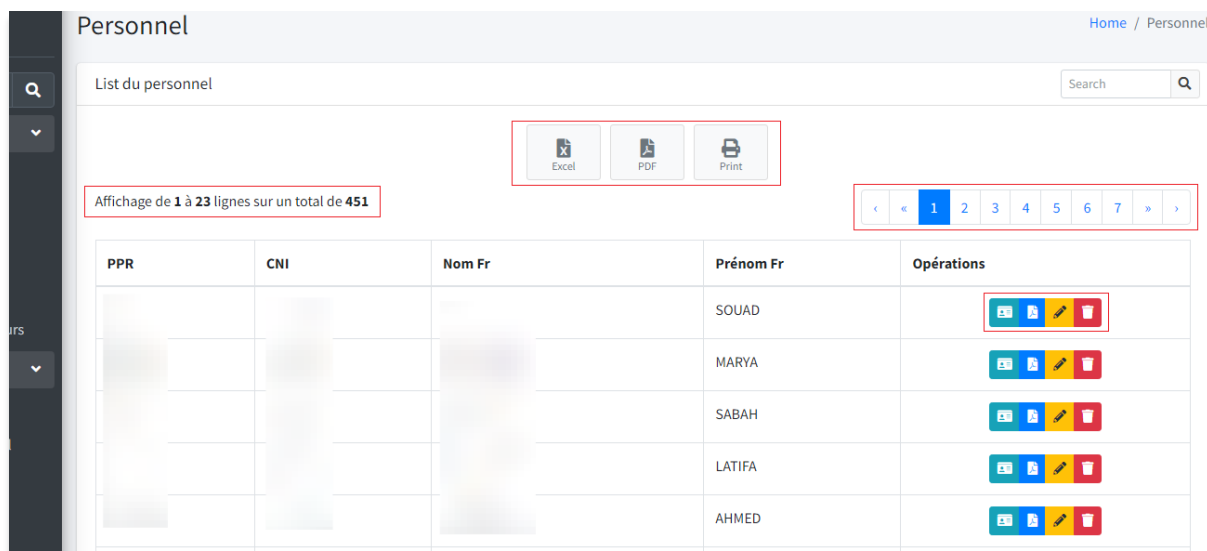



Figure 21 : Capture d'écran de la page 'Gestion du Personnel'

Le clique sur l'icône  d'une ligne permet l'affichage de l'intégralité des informations du personnel sélectionné.

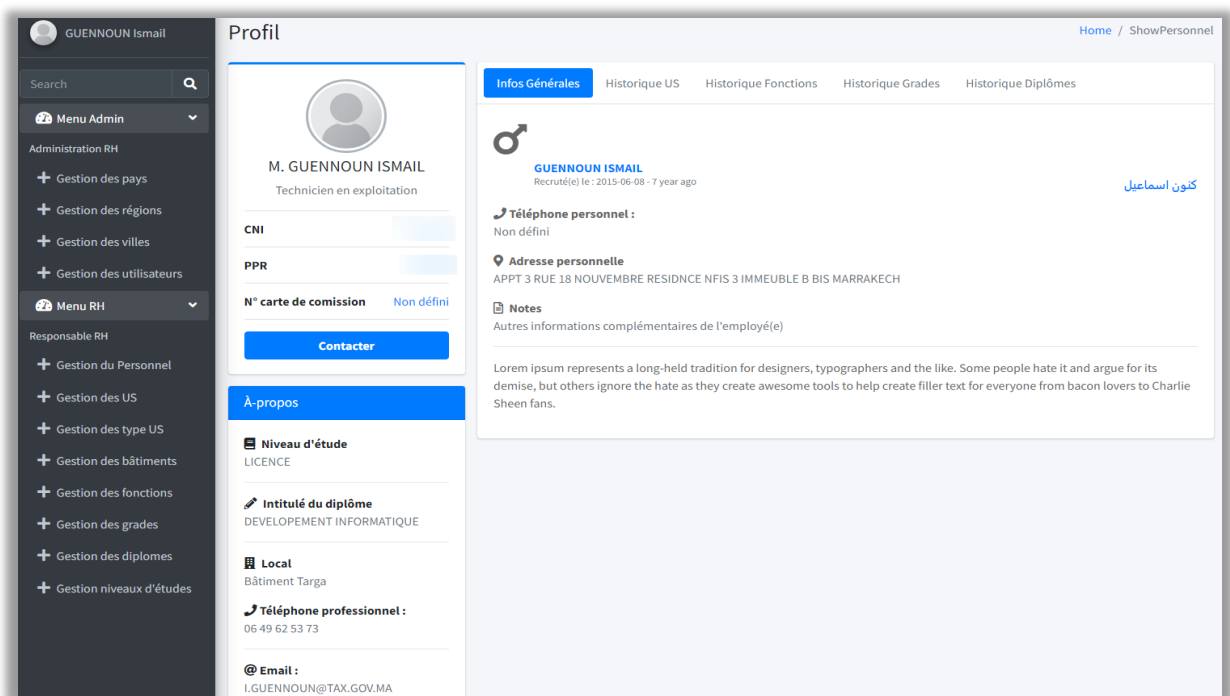


Figure 22 : Capture d'écran de la page 'Profil du personnel'

Le clique sur l'un des onglets d'historique permet d'afficher l'historique des US, Fonctions, grades ou diplôme du personnel.

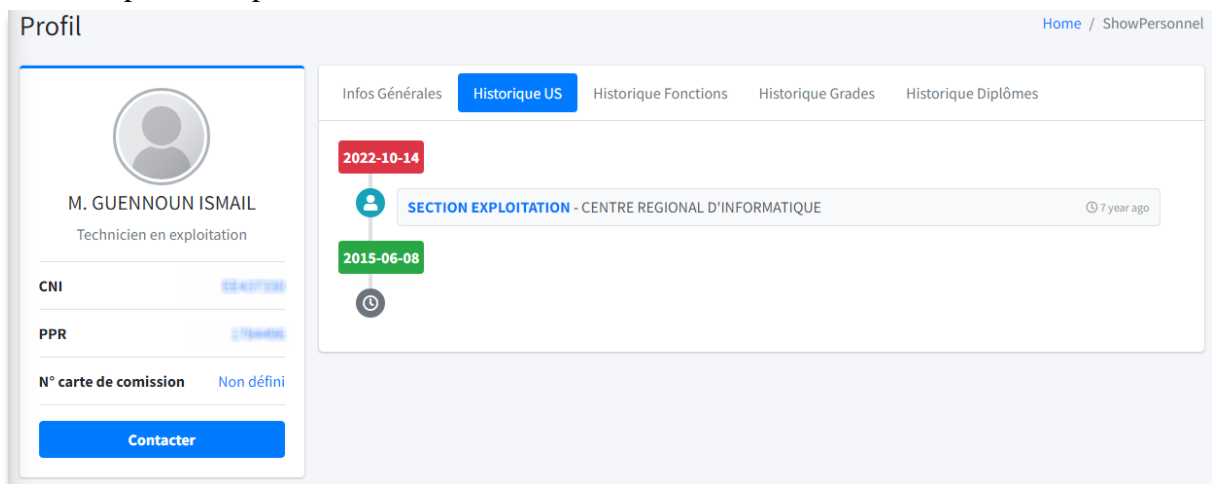


Figure 23 : Capture d'écran de l'onglet 'Historique US' d'un personnel

Pour chacune des gestions disponibles de l'application, un formulaire permet la saisie des informations nécessaires à l'ajout d'enregistrement à la base de donnée. Ci-dessous le formulaire d'ajout d'un personnel :

Formulaire - Personnel

Informations personnelles :

PPR :

3784469

CNI :

EE123456

Le nom en français :

ex : GUENNOUN

Le prénom en français :

ex : Ismail

Le nom en arabe :

ex : كنون

Le prénom en arabe :

ex : اسماعيل

Date de naissance

jj/mm/aaaa

Téléphone personnel :

06 XX XX XX XX

Adresse de résidence :

xx Rue x No xx

Ville de naissance :

Afourar

Sexe :

☒ Homme
 ☐ Femme

Etat civil:

☒ M.
 ☐ Mme.
 ☐ Mlle.

Situation familiale :

☒ Célibataire
 ☐ Marié(e)
 ☐ Divorcé(e)
 ☐ Veuf(ou veuve)

Informations professionnels:

Date de recrutement :

jj/mm/aaaa

Téléphone professionnel :

06 XX XX XX XX ou 05 XX XX XX XX

Email administratif :

p.nom@tax.gov.ma

Numéro de la carte de commission :

4321

Echelle :

9

niveau d'étude :

_BAC

Agent communal :

☐

Détaché(e) d'un autre établissement :

☐

Bâtiment d'affectation :

Bâtiment Targa

Photo :

Figure 24 : Capture d'écran du formulaire d'ajout d'un personnel

Dans ce formulaire, les listes déroulantes (ville de naissance, niveau d'étude, bâtiment d'affectation) sont chargées depuis les table de la BD. Les champs sont regroupés selon le type d'information (personnelle ou professionnel).

V.4.3 Exemples de code

Après le démarrage du serveur depuis le répertoire racine de la partie « Front-end » :

`npm start`

Le serveur démarre et déclare les variables d'environnement dans le fichier '.env' puis exécute le fichier 'index.js' depuis le répertoire '/src', c'est ce fichier qui retourne la page qui sera généré à chaque demande du navigateur et qui appel 'App.js' contenant les routes de l'application web.

```
import { BrowserRouter, Routes, Route } from "react-router-dom";

import Login from "../pages/Login";
import Dashboard from "../pages/Dashboard";
import Personnel from "../pages/Personnel";
import ShowPersonnel from "../pages/ShowPersonnel";
// Ainsi que tous les autres importations des pages web de la partie front-office

function App() {
  return (
    <BrowserRouter>
      <Routes>
        <Route path="/" element={<Login />} />
        <Route path="/dashboard" element={<Dashboard />} />
        <Route path="/personnel/:typeFilter/:codeFilter" element={<Personnel />} />
        <Route path="/personnel/" element={<Personnel />} />
        <Route path="/showpersonnel/:cni" element={<ShowPersonnel />} />
        // Ainsi que tous les autres déclaration des routes utilisées
      </Routes>
    </BrowserRouter>
  );
}

export default App;
```

Figure 25 : Code source - app.js

Le clique sur un élément du menu suscite la génération de la page voulu depuis le répertoire '/pages' qui fait appel aux autres fragment qui la compose depuis '/components'.

Ci-après le squelette d'une page de base de l'application :

```

import Navbar from "../components/Navbar";
import MainSidebar from "../components/MainSidebar";
import MainFooter from "../components/MainFooter";

import axios from "axios";
import {useRef, useEffect, useState } from "react";

const Dashboard = () =>{
  const effectRan = useRef(false);
  const [stat, setStat] = useState({});
  const HOST = process.env.REACT_APP_HOST_URL;
  const PORT = process.env.REACT_APP_HOST_PORT;
  const STAT_URL = HOST+":"+PORT+process.env.REACT_APP_STAT;

  useEffect(()=>{
    if(effectRan.current ===false){
      axios.get(STAT_URL)
        .then(result=>{
          // Récupération des données depuis l'API
          setStat(result.data);
        })
        .catch(error=>{
          console.log(error.message);
        });
    }
    return () => {
      effectRan.current = true
    }
  });
  return (<>
    <div className="wrapper" >
      <Navbar />
      <MainSidebar />
      <div className="content-wrapper">
        <div className="content-header">
          <!-- L'entête du contenu de la page 'tableau de bord' -->
        </div>

        <section className="content">
          <!--Le coprs du contenu de la page 'tableau de bord' -->
        </section>

      </div>
      <MainFooter />
    </div>
  </>)
}
export default Dashboard;

```

Figure 26 : Code source - Squelette d'une page React

La page 'US.js' nécessite un arrêt spécial puisqu'elle utilise un fragment réflexif pour l'affichage de l'arborescence des unités structurales, car elles sont tous dans la même table SQL et que pour chaque US l'affichage diffère selon le fait d'avoir des US enfants ou pas.

```
function IsParent (codeUS){
  let test = false
  // .find or some
  us?.map(one_us =>{
    if(one_us.codeUSParent === codeUS){
      test = true
    }
  })
  return test
};

const USChilds = (codeUSParent) => {
  const USChildArray = []
  us?.map(one_us =>{
    if(one_us.codeUSParent === codeUSParent){
      USChildArray.push(one_us)
    }
  })
  return USChildArray
};

const DrawTree = props =>{
  const {one_us} = props
  if(one_us !== undefined){
    return (
      (!IsParent(one_us.codeUS))?(
        <tr>td className="border-0">{one_us.libelleUSFr}</td></tr>
      ):(
        <Fragment>
        <tr data-widget="expandable-table" aria-expanded="false">
          <td>
            {one_us.libelleUSFr}
          </td>
        </tr>
        <tr className="expandable-body d-none">
          <td><div className="p-0">
            <table className="table table-hover">
              <tbody>{USChilds(one_us.codeUS)?.map(child=>(
                <DrawTree key={child.codeUS} one_us={child}/>
              ))}
            </tbody>
            </table>
          </div></td>
        </tr>
        </Fragment>
      )
    );
  }
};
```

Figure 27 : Code source - Fragment réflexif

V.5 Conclusion

Dans ce chapitre nous avons détaillé les technologies et les outils de développement utilisés ainsi que les fonctionnalités de base de l'application, que ce soit la partie Back-end (API) ou la partie Front-End (Application Web) à travers un ensemble de capture d'écran.

Conclusion Générale

Au cours de ce stage de fin d'étude effectué au sein de la Direction Régional des Impôts de Marrakech, il m'a été possible de mettre en pratique la majorité des connaissances acquis au cours de l'année d'étude de la licence professionnel « Administration de Base de Données » à la Faculté des Sciences d'El Jadida. Autour d'un projet réel couvrant la gestion du personnel de cette administration, j'ai été amené à collaborer avec plusieurs personnes afin de collecter le maximum d'information sur la structure de cette dernière, et sur les différentes taches du bureau des ressources humaines.

C'est dans ce cadre que s'inscrit le présent manuscrit, qui synthétise toutes les étapes par lesquelles ce travail est passé pour arriver au résultat obtenu. Commenant par l'idée du projet jusqu'à l'implémentation final, en passant par l'étude de l'existant et de l'environnement technique, l'analyse des besoins, la conception et la modélisation et en fin le choix de l'intégration des technologies utilisées avec ceux de dernières générations.

Sur le plan technique, nous avons opté pour une architecture trois tiers pour intégrer notre solution à l'environnement disponible. La base de données a été hébergée sur un serveur MySQL, tandis que la partie Back-end coexiste sur un serveur Windows et la partie Front-end installé sur le post du responsable du bureau RH.

Durant la réalisation du projet, nous avons été confrontés à plusieurs problèmes et obstacles, surtout au niveau de la structuration et la consolidation des données de la base de données et du développement de certaines fonctionnalités exigées par les gestionnaires du bureau RH.

Pour étendre notre solution nous pouvant l'enrichir par plus de fonctionnalités (ex : suivie des fournitures). Comme perspectives de travaux futures, la modularité avec laquelle l'application a été développée permet l'ajout d'autre rubriques par ex : gestion de formation, gestion matériel ou toute autre gestion en relation avec le personnel.

Annexe A

Colonnes du fichier « Excel » du traitement initial

Colonnes	Exemples de valeurs
PPR	1234567
NCIN	EE654321
NOMS PRENOMS	GUENNOUN ISMAIL
DOMICILE	Marrakech
DATNAIS	29/08/1991
NAI	1991
LIEU	MARRAKECH
Age	31
Sexe	M
Sexe2	M
Sit famille	C
Fonction	Technicien en exploitation
ABR	MS
Service	CENTRE REGIONAL D'INFORMATIQUE
AB	RH
SECTEUR	SECTION EXPLOITATION
DATREC	08/06/2015
DATE DE MISE A LA DIPOSITION	
DATE DE DETACHEMENT	
GRADE	TECHNICIEN DE 3EME GRADE
Eche	9
OPTION	DEVELOPEMENT INFORMATIQUE
DIPLÔME	LICENCE
ADRESSE	TARGA
N° CARTE COMMISSION	
الإسم	كنون اسماعيل
الدرجة	تقني من الدرجة الثالثة
المصلحة	المركز الجهوي للإعلاميات - المصلحة الجهوية للموارد و النظام المعلوماتي
المكان	تاركة مراكش
ADRESSE PERSONNELLE	Adresse SIS MARRAKECH
TEL	06 49 62 53 73
ADRESSE EMAIL PROFESSIONNELLE	I.GUENNOUN@TAX.GOV.MA

Tableau 6 : Colonnes du fichier « Excel » du traitement initial

Liste des figures

Figure 1 : Organigramme de la DRI de Marrakech.....	4
Figure 2 : Schéma de la méthode XP d'Agile.....	9
Figure 3 : Diagramme de Gantt du Planning.....	11
Figure 4 : Diagramme de cas d'utilisation de l'acteur "Responsable RH"	20
Figure 5 : Diagramme de cas d'utilisation global de l'application.....	21
Figure 6 : Diagramme de séquence du cas d'utilisation "Authentification"	22
Figure 7 : Diagramme de séquence du cas d'utilisation "Ajouter Personnel"	23
Figure 8 : Diagramme de classes de l'application	24
Figure 9 : Schéma de la base de données	25
Figure 10 : Capture d'écran des points de terminaison du Service "Personnel" de L'API	31
Figure 11 : Capture d'écran du point de terminaison - Afficher tous le personnel actifs.....	32
Figure 12 : Capture d'écran du résultat de "Get Actif Personnels"	32
Figure 13 : Code source - main.py	33
Figure 14 : Code source - /routers/personnel.py.....	34
Figure 15 : Requête SQL 'Nombre de personnel par grade'	35
Figure 16 : Requête SQL 'Nombre de personnel prêt de la retraite'	35
Figure 17 : Capture d'écran de la page d'authentification	37
Figure 18 : Capture d'écran des différents Menu selon le profil	37
Figure 19 : Capture d'écran de la page du tableau de bord	38
Figure 20 : Capture d'écran de la page 'Gestion des US'	39
Figure 21 : Capture d'écran de la page 'Gestion du Personnel'	39
Figure 22 : Capture d'écran de la page 'Profil du personnel'.....	40
Figure 23 : Capture d'écran de l'onglet 'Historique US' d'un personnel	40
Figure 24 : Capture d'écran du formulaire d'ajout d'un personnel	41
Figure 25 : Code source - app.js.....	42
Figure 26 : Code source - Squelette d'une page React	43
Figure 27 : Code source - Fragment réflexif	44

Liste des tableaux

Tableau 1 : Etude comparative des 5 solutions de gestion du personnel du marché 2022	15
Tableau 2 : Les acteur et leurs rôles dans le système	17
Tableau 3 : Langages de programmation, Serveurs et Framework utilisés	27
Tableau 4 : Arborescence de la partie "Back-end"	30
Tableau 5 : Arborescence de la partie "Front-end"	36
Tableau 6 : Colonnes du fichier « Excel » du traitement initial	47

Liste des abréviations

API	Interface de Programmation d'Application
BD	Base de donnée
BPM	Business Process Model
CDM	Conceptual Data Model
CRI	Centre Régional Informatique
CRUD	Create, Read, Update, Delete
CSS	Cascading Style Sheets
DGI	Direction Général des Impôts
DMZ	DMmilitarized Zone
DRI	Direction Régional des Impôts
HTML	HyperText Markup Language
HTTP	Hypertext Transfer Protocol
MySQL	My Structured Query Language
NPM	Node Package Manager
PDF	Portable Document Format
RH	Ressources Humaines
RTD	Regional Tax Department
SAAS	Software As A Service
SQL	Structured Query Language
UML	Unified Modeling Language
XP	Extreme Programming

Glossaire des termes techniques

API ⁽¹⁾: En informatique, une interface de programmation d'application ou interface de programmation applicative (souvent désignée par le terme API pour Application Programming Interface) est un ensemble normalisé de classes, de méthodes, de fonctions et de constantes qui sert de façade par laquelle un logiciel offre des services à d'autres logiciels. Elle est offerte par une bibliothèque logicielle ou un service web, le plus souvent accompagnée d'une description qui spécifie comment des programmes consommateurs peuvent se servir des fonctionnalités du programme fournisseur.

ASP ⁽¹²⁾: Est un Framework permettant de générer à la demande des pages web, lancé par Microsoft en juillet 20002, et utilisé pour mettre en œuvre des applications web. Il s'agit d'une évolution d'**Active Server Pages** (ASP, alias Classic ASP), par laquelle cette technique a été incorporée dans la plateforme Microsoft .NET.

Back-end ⁽²⁾: (parfois aussi appelé un **dorsal**) En informatique, un back-end est un terme désignant un étage de sortie d'un logiciel devant produire un résultat. On l'oppose au front-end (aussi appelé un frontal) qui lui est la partie visible de l'iceberg.

Bibliothèque logicielle ⁽²⁰⁾: En informatique, une bibliothèque ou librairie logicielle (ou encore, bibliothèque de programmes) est un ensemble de fonctions utilitaires, regroupées et mises à disposition afin de pouvoir être utilisées sans avoir à les réécrire. Les fonctions sont regroupées de par leur appartenance à un même domaine conceptuel (mathématique, graphique, tris, etc). Les bibliothèques logicielles se distinguent des exécutables dans la mesure où elles ne représentent pas une application.

Désérialiser ⁽³¹⁾ : est un anglicisme qui consiste à rassembler des informations diverses en un tout. Extraire des données d'un flux et les convertir dans un format utile à l'application.

DMZ ⁽¹⁰⁾: En informatique, une zone démilitarisée, ou DMZ (en anglais, **demilitarized zone**) est un sous-réseau séparé du réseau local et isolé de celui-ci ainsi que d'Internet (ou d'un autre réseau) par un pare-feu. Ce sous-réseau contient les machines étant susceptibles d'être accédées depuis Internet, et qui n'ont pas besoin d'accéder au réseau local.

Front-end ⁽³⁴⁾: ou (**Web frontal** en français) correspond aux productions HTML, CSS et JavaScript d'une page internet ou d'une application qu'un utilisateur peut voir et avec lesquelles il peut interagir directement.

Framework ⁽³⁵⁾: appelé aussi infrastructure logicielle, est un ensemble cohérent de composants logiciels structurels qui sert à créer les fondations ainsi que les grandes lignes de tout ou partie d'un logiciel, l'ensemble formant ou promouvant un « squelette » de programme, un canevas. En programmation orientée objet, un framework est typiquement composé de classes mères qui seront dérivées et étendues par héritage en fonction des besoins spécifiques à chaque logiciel qui l'utilise.

HTTP ⁽³⁶⁾: littéralement « protocole de transfert hypertexte », est un protocole de communication client-serveur développé pour le World Wide Web. Dans le protocole HTTP, une méthode est une commande spécifiant un type de requête, c'est-à-dire qu'elle demande au serveur d'effectuer une action. En général l'action concerne une ressource identifiée par l'URL qui suit le nom de la méthode.

IIS ⁽¹¹⁾: Internet Information Services, anciennement **Internet Information Server**, communément appelé IIS (prononcé généralement "2 i s"), est un serveur Web (HTTP) des différents systèmes d'exploitation Windows NT.

INGRES ⁽¹⁴⁾: Ingres ou Actian est un système de gestion de base de données (SGBD) relationnel. Ingres signifie : INtelligent GRaphic RELational System.

Son concepteur, Michael Stonebraker, décida de recommencer le développement à partir de zéro en 1985, mais en continuant à développer les idées d'Ingres. Il se lança dans un projet post-Ingres, qui fut d'abord nommé Postgres, puis, à partir de 1995, PostgreSQL.

INTERNET ⁽³⁷⁾: Est un réseau informatique mondial accessible au public. Il s'agit d'un réseau de réseaux, à commutation de paquets, sans centre névralgique, composé de millions de réseaux aussi bien publics que privés, universitaires, commerciaux et gouvernementaux, eux-mêmes regroupés en réseaux autonomes.

JIRA ⁽³⁸⁾: Est un système de suivi de bugs, de gestion des incidents et de gestion de projets développé par **Atlassian** et publié pour la première fois en 2002. Il propose des solutions à la fois à destination des développeurs et des intervenants non développeurs.

MEGAFORM ⁽¹⁵⁾: est une solution très simple et efficace qui permet de remplacer les formulaires préimprimés et les anciennes imprimantes matricielles par des documents de haute qualité imprimés avec des imprimantes laser.

NPM ⁽³⁹⁾: est le gestionnaire de paquets par défaut pour l'environnement d'exécution JavaScript de **Node.js**.

OpenAPI ⁽⁴⁰⁾: La spécification OpenAPI, anciennement connue sous le nom de spécification Swagger, est une spécification pour les fichiers d'interface lisibles par machine pour décrire, produire, consommer et visualiser des services Web RESTful

PDF ⁽⁴¹⁾: **Le Portable Document Format**, communément abrégé en PDF, est un langage de description de page présenté par la société Adobe Systems en 1992 et qui est devenu une norme ISO en 2008. La spécificité du PDF est de préserver la mise en page d'un document « polices de caractères, images, objets graphiques, etc. » telle qu'elle a été définie par son auteur, et cela quels que soient le logiciel, le système d'exploitation et l'ordinateur utilisés pour l'imprimer ou le visualiser.

Pydantic ⁽²⁹⁾: Bibliothèque Python pour l'analyse et la validation de données à l'aide d'indications de type Python.

RESTful ⁽²⁸⁾: **Representational State Transfer** est un style d'architecture logicielle définissant un ensemble de contraintes à utiliser pour créer des services web.

SAAS ⁽¹⁹⁾: Le **Software As A Service** ou logiciel en tant que service, est un modèle d'exploitation commerciale des logiciels dans lequel ceux-ci sont installés sur des serveurs distants plutôt que sur la machine de l'utilisateur. Les clients ne paient pas de licence d'utilisation pour une version, mais utilisent librement le service en ligne ou, plus généralement, payent un abonnement.

Sérialisation ⁽³⁰⁾: est le codage d'une information sous la forme d'une suite d'informations plus petites (dites atomiques) pour, par exemple, sa sauvegarde (persistance) ou son transport sur le réseau

SLACK ⁽⁴²⁾: Est une plateforme de communication collaborative propriétaire (SaaS) ainsi qu'un logiciel de gestion de projets. Slack est l'acronyme de « **Searchable Log of All Conversation and Knowledge** », qui peut se traduire en français, "journal de bord avec recherche de toutes les conversations et connaissances" ou "Accessibilité à toutes les conversations et connaissances".

TEAMS ⁽⁴³⁾: ou **Microsoft Teams** est une application de communication collaborative propriétaire en mode SaaS officiellement lancée par Microsoft en novembre 2016. Le service s'intègre à la suite Microsoft 365 en remplacement de Skype for Business et propose des extensions pouvant être intégrées à des produits autres que ceux de Microsoft.

Template ⁽⁴³⁾: connu également sous le terme de modèle ou Gabarit en français, représente l'ensemble des éléments graphiques de l'agencement des colonnes, passant par le choix des couleurs jusqu'à l'établissement de la structure des différents éléments enveloppant un site Internet, abstraction faite de son contenu.

Webographie

1. INTERFACE DE PROGRAMMATION. [EN LIGNE] [CITATION : 26 08 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/INTERFACE_DE_PROGRAMMATION.](https://fr.wikipedia.org/wiki/Interface_de_programmation)
2. BACKEND. WIKIPEDIA. [EN LIGNE] [CITATION : 29 09 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/BACKEND.](https://fr.wikipedia.org/wiki/Backend)
3. LA DIRECTION GENERALE DES IMPOTS. [EN LIGNE] [CITATION : 26 08 2022.]
[HTTPS://WWW.FINANCES.GOV.MA/FR/MINISTERE/PAGES/DGI.ASPX.](https://www.finances.gov.ma/fr/ministere/pages/dgi.aspx)
4. ORGANISATION DE LA DGI. [EN LIGNE] [CITATION : 26 08 2022.]
[HTTPS://TAX.GOV.MA/WPS/PORTAL/DGI/NOUS-CONNAITRE/NOTRE-ORGANISATION.](https://tax.gov.ma/wps/portal/dgi/nous-connaître/notre-organisation)
5. VALEURS DE LA DGI. [EN LIGNE] [CITATION : 26 08 2022.]
[HTTPS://TAX.GOV.MA/WPS/PORTAL/DGI/NOUS-CONNAITRE/NOS-VALEURS.](https://tax.gov.ma/wps/portal/dgi/nous-connaître/nos-valeurs)
6. MISSION DE LA DGI. [EN LIGNE] [CITATION : 26 08 2022.]
[HTTPS://TAX.GOV.MA/WPS/PORTAL/DGI/NOUS-CONNAITRE/NOS-MISSIONS-ET-ATTRIBUTIONS.](https://tax.gov.ma/wps/portal/dgi/nous-connaître/nos-missions-et-attributions)
7. VISION STRATEGIQUE DE LA DGI. [EN LIGNE] [CITATION : 26 08 2022.]
[HTTPS://TAX.GOV.MA/WPS/PORTAL/DGI/NOUS-CONNAITRE/OBJECTIFS-STRATEGIQUES-DE-LA-DGI.](https://tax.gov.ma/wps/portal/dgi/nous-connaître/objectifs-strategiques-de-la-dgi)
8. QU'EST CE QUE LA METHODOLOGIE EXTREME PROGRAMMING. PLANZONE.FR. [EN LIGNE]
[CITATION : 27 08 2022.] [HTTPS://WWW.PLANZONE.FR/BLOG/QUEST-CE-QUE-LA-METHODOLOGIE-EXTREME-PROGRAMMING.](https://www.planzone.fr/blog/quest-ce-que-la-methodologie-extreme-programming)
9. GANTTPROJECT. WIKIPEDIA. [EN LIGNE] [CITATION : 27 08 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/GANTTPROJECT.](https://fr.wikipedia.org/wiki/GanttProject)
10. ZONE DEMILITARISEE (INFORMATIQUE). WIKIPEDIA. [EN LIGNE] [CITATION : 28 08 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/ZONE_D%C3%A9MILITARIS%C3%A9_\(INFORMATIQUE\).](https://fr.wikipedia.org/wiki/Zone_d%27%C3%A9militarisation_(informatique))
11. INTERNET INFORMATION SERVICES. WIKIPEDIA. [EN LIGNE] [CITATION : 28 08 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/INTERNET_INFORMATION_SERVICES.](https://fr.wikipedia.org/wiki/Internet_information_services)
12. ASP.NET. WIKIPEDIA. [EN LIGNE] [CITATION : 28 08 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/ASP.NET.](https://fr.wikipedia.org/wiki/ASP.NET)
13. MYSQL. WIKIPEDIA. [EN LIGNE] [CITATION : 28 08 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/MYSQL.](https://fr.wikipedia.org/wiki/MySQL)

14. INGRES(BASE DE DONNEES). WIKIPEDIA. [EN LIGNE] [CITATION : 28 08 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/INGRES_\(BASE_DE_DONN%C3%A9ES\).](https://fr.wikipedia.org/wiki/INGRES_(BASE_DE_DONN%C3%A9ES))
15. MEGAFORM. OBERON. [EN LIGNE] [CITATION : 28 08 2022.]
[HTTP://WWW.OBERON.IT/EN/PRODUCTS/MEGAFORM.HTML.](http://www.oberon.it/en/products/megaform.html)
16. 5 LOGICIELS DE GESTION DES RESSOURCES HUMAINES. ZDNET. [EN LIGNE] [CITATION : 28 08 2022.] [HTTPS://WWW.ZDNET.FR/GUIDE-ACHAT/5-LOGICIELS-DE-GESTION-DES-RESSOURCES-HUMAINES-39930767.HTM.](https://www.zdnet.fr/guide-achat/5-logiciels-de-gestion-des-ressources-humaines-39930767.htm)
17. UML (INFORMATIQUE). WIKIPEDIA. [EN LIGNE] [CITATION : 29 08 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/UML_\(INFORMATIQUE\).](https://fr.wikipedia.org/wiki/UML_(informatique))
18. POWERAMC. WIKIPEDIA. [EN LIGNE] [CITATION : 29 08 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/POWERAMC.](https://fr.wikipedia.org/wiki/PowerAMC)
19. SOFTWARE AS A SERVICE. WIKIPEDIA. [EN LIGNE] [CITATION : 30 08 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/SOFTWARE_AS_A_SERVICE.](https://fr.wikipedia.org/wiki/Software_as_a_service)
20. BIBLIOTHEQUE LGICIELLE. TECHNO-SCIENCE. [EN LIGNE] [CITATION : 30 09 2022.]
[HTTPS://WWW.TECHNO-SCIENCE.NET/DEFINITION/1470.HTML.](https://www.techno-science.net/definition/1470.html)
21. JAVASCRIPT. JAVASCRIPT. [EN LIGNE] [CITATION : 30 09 2022.]
[HTTPS://WWW.W3SCHOOLS.COM/JS/.](https://www.w3schools.com/js/)
22. REACT. REACTJS. [EN LIGNE] [CITATION : 30 09 2022.] [HTTPS://FR.REACTJS.ORG/.](https://fr.reactjs.org/)
23. PYTHON (LANGAGE). WIKIPEDIA. [EN LIGNE] [CITATION : 30 09 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/PYTHON_\(LANGAGE\).](https://fr.wikipedia.org/wiki/Python_(langage))
24. HTML. DEVELOPER.MOZILLA. [EN LIGNE] [CITATION : 30 09 2022.]
[HTTPS://DEVELOPER.MOZILLA.ORG/FR/DOCS/WEB/HTML.](https://developer.mozilla.org/fr/docs/Web/HTML)
25. CSS. W3SCHOLLS. [EN LIGNE] [CITATION : 30 09 2022.] [HTTPS://WWW.W3SCHOOLS.COM/CSS/.](https://www.w3schools.com/css/)
26. ADMINLTE. ADMINLTE. [EN LIGNE] [CITATION : 30 09 2022.] [HTTPS://ADMINLTE.IO/.](https://adminlte.io/)
27. FASTAPI. WIKIPEDIA. [EN LIGNE] [CITATION : 30 09 2022.]
[HTTPS://EN.WIKIPEDIA.ORG/WIKI/FASTAPI.](https://en.wikipedia.org/wiki/FastAPI)
28. REPRESENTATIONAL STATE TRANSFER. WIKIPEDIA. [EN LIGNE] [CITATION : 01 10 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/REPRESENTATIONAL_STATE_TRANSFER.](https://fr.wikipedia.org/wiki/Representational_state_transfer)
29. PYDANTIC. WIKIDATA. [EN LIGNE] [CITATION : 01 10 2022.]
[HTTPS://WWW.WIKIDATA.ORG/WIKI/Q107381687.](https://www.wikidata.org/wiki/Q107381687)
30. SERIALISATION. WIKIPEDIA. [EN LIGNE] [CITATION : 01 10 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/S%C3%A9RIALISATION.](https://fr.wikipedia.org/wiki/S%C3%A9rialisation)

31. DÉSÉRIALISER . WIKTIONARY. [EN LIGNE] [CITATION : 01 10 2022.]
[HTTPS://FR.WIKTIONARY.ORG/WIKI/D%C3%A9s%C3%A9rialiser](https://fr.wiktionary.org/wiki/d%C3%A9s%C3%A9rialiser).
32. VISUAL STUDIO CODE. WIKIPEDIA. [EN LIGNE] [CITATION : 01 10 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/VISUAL_STUDIO_CODE](https://fr.wikipedia.org/wiki/Visual_Studio_Code).
33. GIT. WIKIPEDIA. [EN LIGNE] [CITATION : 01 10 2022.] [HTTPS://FR.WIKIPEDIA.ORG/WIKI/GIT](https://fr.wikipedia.org/wiki/Git).
34. DEVELOPPEMENT WEB FRONTAL. WIKIPEDIA. [EN LIGNE] [CITATION : 29 09 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/D%C3%A9veloppement_web_frontal](https://fr.wikipedia.org/wiki/D%C3%A9veloppement_web_frontal).
35. FRAMEWORK. WIKIPEDIA. [EN LIGNE] [CITATION : 30 09 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/FRAMEWORK](https://fr.wikipedia.org/wiki/Framework).
36. HYPERTEXT TRANSFER PROTOCOL. WIKIPEDIA. [EN LIGNE] [CITATION : 30 09 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/HYPERTEXT_TRANSFER_PROTOCOL](https://fr.wikipedia.org/wiki/Hypertext_transfer_protocol).
37. INTERNET. WIKIPEDIA. [EN LIGNE] [CITATION : 30 08 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/INTERNET](https://fr.wikipedia.org/wiki/Internet).
38. JIRA. WIKIPEDIA. [EN LIGNE] [CITATION : 30 08 2022.] [HTTPS://FR.WIKIPEDIA.ORG/WIKI/JIRA](https://fr.wikipedia.org/wiki/Jira).
39. NPM. WIKIPEDIA. [EN LIGNE] [CITATION : 30 09 2022.] [HTTPS://FR.WIKIPEDIA.ORG/WIKI/NPM](https://fr.wikipedia.org/wiki/Npm).
40. OPENAPI SPECIFICATION. WIKIPEDIA. [EN LIGNE] [CITATION : 01 10 2022.]
[HTTPS://EN.WIKIPEDIA.ORG/WIKI/OPENAPI_SPECIFICATION](https://en.wikipedia.org/wiki/OpenAPI_specification).
41. PDF. WIKIPEDIA. [EN LIGNE] [CITATION : 30 08 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/PORTABLE_DOCUMENT_FORMAT](https://fr.wikipedia.org/wiki/Portable_document_format).
42. SLACK (PLATFOME). WIKIPEDIA. [EN LIGNE] [CITATION : 30 08 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/SLACK_\(PLATEFOME\)](https://fr.wikipedia.org/wiki/Slack_(plateforme)).
43. TEMPLATE (PAGE WEB). TAKTILCOMMUNICATION. [EN LIGNE] [CITATION : 30 09 2022.]
[HTTPS://WWW.TAKTILCOMMUNICATION.COM/GLOSSAIRE/STUDIO-GRAPHIQUE-ET-MULTIMEDIA/TEMPLATE.HTML](https://www.taktilcommunication.com/glossaire/studio-graphique-et-multimedia/template.html).
44. MICROSOFT TEAMS. WIKIPEDIA. [EN LIGNE] [CITATION : 30 08 2022.]
[HTTPS://FR.WIKIPEDIA.ORG/WIKI/MICROSOFT_TEAMS](https://fr.wikipedia.org/wiki/Microsoft_Teams).