



Université de la Manouba
École Nationale des Sciences de l'Informatique



Rapport du Projet de Conception et de Développement

**Sujet : Using Data Mining to sort out the
productivity trend for each developers in
Agile projects.**

Auteurs :

M. Khalil GORSAN MESTIRI M. Seif DASSI

Encadrants :

Mme. Nesrine BEN YAHIA
M. Housseem BELKHOUBA

Organisme :



Année Universitaire :2017 /2018

Appréciation et signature de l'encadrant

MME. NESRINE BEN YAHIA



Remerciements

Nous tenons à exprimer nos vifs remerciements et notre profonde gratitude à toute personne ayant contribué de près ou de loin à la réalisation de notre projet de conception et développement :

Nous remercions énormément Mme.**Nesrine Ben Yahia** de nous avoir encadrés durant la réalisation de notre projet.

Nous tenons, également, à remercier M.**Belkhouja Housseem** de nous avoir, tout d'abord, proposés un sujet à la fois intéressant et instructif, et de nous avoir encadrés au sein de l'entreprise Fidelity National Information Services.

Ce fut un énorme plaisir de travailler à vos côtés.



Table des matières

| | |
|--|----------|
| Introduction générale | 1 |
| 1 Présentation générale du projet et Etat de l'art | 3 |
| Introduction | 3 |
| 1.1 Présentation générale du projet | 3 |
| 1.1.1 Cadre de projet | 3 |
| 1.1.2 Présentation de l'organisme d'accueil | 3 |
| 1.1.2.1 Fidelity National Information Services : | 3 |
| 1.1.2.2 ENSI Junior Entreprise | 4 |
| 1.1.3 Méthode agile | 4 |
| 1.1.3.1 Méthode Scrum | 4 |
| 1.1.4 Travail demandé | 5 |
| 1.2 Etat de l'art | 5 |
| 1.2.1 Métier de ressource humaine | 5 |
| 1.2.1.1 Définition | 5 |
| 1.2.1.2 Classification des méthodes de gestion de ressource hu- maine | 6 |
| 1.2.1.3 Les personnels | 6 |
| 1.2.1.4 Technologie et RH | 7 |
| 1.2.2 Étude de l'existant | 7 |
| 1.2.2.1 IBM Watson | 7 |
| 1.2.2.2 SP-Expert | 7 |
| 1.2.2.3 Critiques de l'existant | 8 |
| 1.2.3 Solution proposée | 8 |
| 1.2.3.1 Définitions | 8 |
| 1.2.3.2 Utilisation de l'IA dans notre application | 9 |
| 1.2.3.3 La valeur ajoutée | 9 |

| | |
|--|-----------|
| Conclusion | 9 |
| 2 Analyse des besoins | 10 |
| Introduction | 10 |
| 2.1 Spécification des besoins | 10 |
| 2.1.1 Identification des acteurs | 10 |
| 2.1.2 Les besoins fonctionnels | 10 |
| 2.1.3 Les besoins non fonctionnels | 11 |
| 2.1.4 Les Besoins du domaine | 11 |
| 2.2 Modélisation | 12 |
| 2.2.1 Langage de modélisation | 12 |
| 2.2.2 Diagramme des cas d'utilisation | 12 |
| 2.2.3 Diagrammes de séquences | 13 |
| 2.2.3.1 Diagramme de séquences illustrant le scénario : "Ajouter un Employée" | 14 |
| 2.2.3.2 Diagramme de séquences illustrant le scénario : "Visualiser indicateurs de performance" | 15 |
| Conclusion | 15 |
| 3 Conception | 17 |
| Introduction | 17 |
| 3.1 Conception générale | 17 |
| 3.1.1 Architecture physique | 18 |
| 3.1.2 Architecture logique | 19 |
| 3.1.3 Choix de l'architecture | 20 |
| 3.1.4 Le diagramme de paquetage | 21 |
| 3.1.4.1 Description des paquetages | 21 |
| 3.2 Conception détaillée | 22 |
| 3.2.1 Conception de la base de données | 22 |
| 3.2.1.1 Modèle Entité-Association | 22 |
| 3.2.1.2 Identification des entités | 22 |
| 3.2.1.3 Identification des associations | 23 |
| 3.2.1.4 Modèle logique des données | 24 |
| 3.2.2 Diagrammes de séquences d'objets | 24 |

| | | |
|----------|--|-----------|
| 3.2.2.1 | Diagramme de séquences objets «Ajouter un employé» . . . | 24 |
| 3.2.2.2 | Diagramme de séquences objets « Visualiser indicateurs de performance » | 26 |
| | Conclusion | 27 |
| 4 | Réalisation | 28 |
| | Introduction | 28 |
| 4.1 | Environnement de travail | 28 |
| 4.1.1 | Environnement matériel | 28 |
| 4.1.2 | Environnement logiciel | 29 |
| 4.1.3 | Choix technologiques | 30 |
| 4.2 | Explication du travail réalisé | 31 |
| 4.2.1 | Cas de test de prédiction | 31 |
| 4.3 | Aperçu sur le travail réalisé | 35 |
| 4.3.1 | Page d'authentification | 35 |
| 4.3.2 | Page d'accueil | 36 |
| 4.3.3 | Interface d'ajout des employés par un expert | 36 |
| 4.3.4 | Interface d'affichage des employés | 37 |
| 4.3.5 | Interface d'affichage des graphes | 38 |
| 4.4 | Chronogramme de gestion de travail | 39 |
| | Conclusion | 39 |
| | Conclusion générale et perspectives | 40 |

Table des figures

| | | |
|-----|--|----|
| 2.1 | Diagramme de cas d'utilisation du système | 13 |
| 2.2 | diagramme de séquences système du cas d'utilisation "Ajouter un Employé" | 14 |
| 2.3 | diagramme de séquences système du cas d'utilisation "Visualiser indicateurs de performance" | 15 |
| 3.1 | Architecture globale de l'application | 17 |
| 3.2 | Architecture trois-tiers | 18 |
| 3.3 | Patron de conception MVT | 20 |
| 3.4 | Diagramme de paquetage | 21 |
| 3.5 | Diagramme Entité-Association | 22 |
| 3.6 | Diagramme de séquences objets « Ajouter un employé » | 25 |
| 3.7 | Diagramme de séquences objets « Visualiser indicateurs de performance » . | 26 |
| 4.1 | Arbre de décision | 32 |
| 4.2 | Parcours de l'Arbre de décision | 34 |
| 4.3 | Interface de login | 35 |
| 4.4 | Interface d'accueil | 36 |
| 4.5 | Interface d'ajout des employés par un expert | 37 |
| 4.6 | Interface d'affichage des indicateurs des employés | 37 |
| 4.7 | Interface graphique illustrant les critères de performance | 38 |
| 4.8 | Chronogramme de gestion de travail | 39 |



Liste des tableaux

| | | |
|-----|---|----|
| 3.1 | Tableau d'identification des entités | 23 |
| 3.2 | Tableau d'identification des associations | 24 |



Introduction générale

L'exploration des données consiste à analyser une grande quantité de données depuis différentes perspectives dans l'optique de transformer ses données en des informations utiles, en établissant des liens et des relations entre ces données. Le résultat tiré s'avère, par la suite, énormément important pour les entreprises puisqu'il contribue directement à l'augmentation de leurs chiffres d'affaires ou la réduction de leurs coûts.

Bien que plusieurs entreprises utilisent l'exploration des données comme un outil d'analyse des comportements et des préférences des consommateurs afin d'établir de meilleures stratégies marketing, d'autres entreprises focalisent cette technique sur la gestion de leurs ressources humaines. En effet, de plus en plus d'outils de « data mining » sont utilisés afin d'identifier les meilleurs employés au sein de l'entreprise.

En collectant suffisamment de données sur ses employés, l'entreprise se voit capable de déterminer le niveau de productivité et de satisfaction de ces derniers. Ce qui facilitera d'une façon considérable l'administration, la mobilisation et le développement des ressources humaines impliquées dans l'activité de l'entreprise. Par conséquent, ces finalités s'avèrent au cœur du développement du monde entrepreneurial, et s'imposent comme un défi incontournable pour tous ceux qui s'intéressent à la gestion interne des entreprises.

Soucieux de l'intérêt majeur derrière une telle problématique, nous avons pris l'initiative de traiter cette éventualité à travers notre projet de conception et de développement. En effet, notre projet consiste à déterminer les niveaux de productivité de chaque développeur dans les méthodologies agiles, en se basant sur l'analyse de leurs données.

Le présent rapport s'articule sur cinq chapitres à travers lesquels nous présenterons le travail réalisé. Le premier chapitre est un chapitre introductif illustrant le contexte général de notre projet : description du cadre de notre projet, présentation de l'organisme d'accueil, énonciation du travail demandé. Le deuxième chapitre, aura pour objectifs : une étude préliminaire qui mettra en valeur les notions théoriques de notre projet, une étude de l'existant accompagnée d'une critique pour aboutir à la définition de la solution proposée. Le troisième chapitre est consacré à une analyse des besoins et à la formalisation

de la spécification du projet ce qui permet de procéder à la représentation de l'interaction de l'utilisateur avec le système. Le quatrième chapitre, apporte une démonstration de la conception générale et détaillée adoptée pour répondre aux besoins précédemment cités. Le dernier chapitre illustrera la partie réalisation dans lequel nous présenterons l'environnement de développement ainsi que l'application de notre travail à travers quelques interfaces de l'application. Enfin, nous clôturerons par une conclusion générale sur le projet effectué tout au long de ce travail ainsi que les perspectives pouvant lui donner d'éventuelles améliorations.

Présentation générale du projet et Etat de l'art

Introduction

Dans ce premier chapitre, nous commençons, pour une première partie, par présenter le cadre de projet. Ensuite, nous présentons l'organisme dans lequel notre projet a été réalisé. Enfin, nous exposons le travail demandé. Dans la deuxième partie, nous présentons le métier de ressource humaine dans une entreprise ainsi que son impact sur les performances des employés. Nous enchaînons par une étude de la solution actuelle ainsi qu'une critique de l'existant pour finir par la solution proposée.

1.1 Présentation générale du projet

1.1.1 Cadre de projet

Ce projet se situe dans le contexte du projet de conception et développement au sein de l'école nationale des sciences de l'informatique. Ce projet est réalisé dans le cadre d'un partenariat entre l'association ENSI Junior Entreprise et l'entreprise « Fidelity National Information Services »

1.1.2 Présentation de l'organisme d'accueil

Afin de situer notre travail dans son environnement de réalisation, nous présentons dans cette section l'organisme d'accueil, en énumérant ses secteurs d'activités.

1.1.2.1 Fidelity National Information Services :

Fidelity National Information Services, plus connu sous l'abréviation « FIS », est un leader mondial dans la technologie des services financiers (axé prioritairement sur les services financiers institutionnel et de détail, les paiements, la gestion d'actifs et de patrimoine, le risque et conformité et les solutions de consulting et d'outsourcing). Ayant son siège à Jacksonville-Floride, FIS est présente dans le classement Fortune 500 et fait

partie de l'indice 500 Standard and Poor's. Elle sert plus de 20 000 Clients dans plus de 130 pays et totalise plus de 55 000 collaborateurs et génère 9,3 milliards de dollars de chiffres d'affaires.[1]

1.1.2.2 ENSI Junior Entreprise

Fondée en 2006, l'ENSI Junior Entreprise est une association étudiante à but non lucratif, inscrite au journal officiel de la république Tunisienne et affiliée à l'école nationale des sciences de l'informatique, ayant pour finalité l'initiation des étudiants tunisiens à la vie professionnelle. Et ce à travers l'élaboration des activités basées sur trois axes principaux : Les formations, les événements et pour finir la réalisation des projets dans le domaine des TIC. Figurant parmi les pionniers du mouvement Junior Entreprise en Tunisie, l'ENSI Junior Entreprise est arrivée à acquérir de l'expérience et de l'expertise, toujours tirée par un large éventail du savoir-faire, des compétences et de l'esprit d'entreprise.[2]

1.1.3 Méthode agile

Une méthode agile est une approche itérative et incrémentale, qui est menée dans un esprit collaboratif. Elle génère un produit de haute qualité tout en prenant en compte l'évolution des besoins des clients. En effet, les méthodes agiles se veulent plus pragmatiques que les méthodes traditionnelles, impliquent au maximum le demandeur ou le client, et permettent une grande réactivité à ses demandes. Elles reposent sur un cycle de développement itératif, incrémental et adaptatif. La méthode agile désormais la plus utilisée est la méthode Scrum.

1.1.3.1 Méthode Scrum

Il s'agit d'un schéma d'organisation de développement de produits complexes. Il est défini par ses créateurs comme un « cadre de travail holistique itératif qui se concentre sur les buts communs en livrant de manière productive et créative des produits de la plus grande valeur possible ». Le cœur de Scrum est un sprint : Un bloc de temps d'un mois ou moins durant lequel un incrément du produit est réalisé. Ces sprints sont le résultat du découpage du projet en boîtes de temps. Chaque sprint commence par une estimation suivie d'une planification opérationnelle. Le sprint se termine par une démonstration de ce qui a été achevé. Avant de démarrer un nouveau sprint, l'équipe réalise une rétrospective. Cette technique analyse le déroulement du sprint achevé, afin d'améliorer ses pratiques. Le flot de travail de l'équipe de développement est facilité par son auto-organisation, il

n'y aura donc pas de gestionnaire de projet.

1.1.4 Travail demandé

Notre travail consiste à développer une application web qui permet de définir le niveau de productivité d'un développeur de l'entreprise. D'une façon plus précise, nous sommes amenés à mettre en place les critères ayant un impact direct ou indirect sur la productivité d'un employé. Ensuite, nous serons dans l'obligation de regrouper les critères en famille et trouver les liens entre eux. Du coup, à l'aide des données enregistrées dans les bases de données de l'entreprise, nous serons capables d'appliquer l'apprentissage automatique à notre modèle. D'autre part, une interface web présentant un tableau de bord de chaque employé doit être mise en œuvre pour offrir la possibilité à chaque employé concerné de consulter son niveau de productivité.

D'une façon plus globale, nous pouvons définir trois tâches majeures à exécuter :

- Dans un premier temps, il est impératif d'identifier, modéliser et classer les critères affectant la productivité d'un développeur au sein d'une entreprise.
- Dans un second temps, la mission sera de trouver les liens reliant les données, afin de constituer un système capable de déduire le niveau de productivité d'un développeur à partir de ses données.
- Enfin, il est nécessaire de présenter ces résultats à travers une application web.

1.2 Etat de l'art

1.2.1 Métier de ressource humaine

Dans cette partie, nous allons définir le métier de ressource humaine et la classification des méthodes afin de garder une performance stable des employés. Nous allons identifier les différents intervenants des entreprises qui se chargent de ce volet et nous finissons par l'identification des solutions technologiques dédiés à ce besoin.

1.2.1.1 Définition

La gestion de ressource humaine est est l'ensemble des pratiques mises en œuvre pour administrer, mobiliser et développer les ressources humaines impliquées dans l'activité d'une organisation. [3]

- Dans un premier temps, cette fonction est entendue dans une perspective opérationnelle. Il s'agit d'administrer un personnel qui peut être numériquement important et

réparti en différents niveaux de hiérarchie ou de qualification.

- Dans un second temps, la fonction acquiert une dimension plus fonctionnelle. Il s'agit d'améliorer la communication transversale entre services et processus et de mettre en œuvre un développement des collaborateurs tout au long de leur séjour dans l'organisation.

1.2.1.2 Classification des méthodes de gestion de ressource humaine

Nous nous intéresserons dans cette partie aux différents processus utilisés pour gérer les ressources humaines.

Analytique RH : C'est le processus qui permet de prendre une décision optimale ou réaliste basée sur les données existantes de l'organisation. On parle donc ici d'appuyer les décisions RH par des faits tangibles et non seulement en s'appuyant sur des opinions ou des intuitions. Par conséquent cela permettra de :

- Déterminer qui est à risque de quitter l'organisation.
- Prédire la performance d'un nouvel employé.
- Déterminer les meilleurs cheminement des carrières.

La reconnaissance : Les entreprises soulignent la précieuse contribution de leurs collaborateurs dans le but de les garder motivés et productifs. Cette stratégie consiste à récompenser les employés que ce soit par une rémunération, une mobilité ou moralement dans l'optique de favoriser l'instauration d'un milieu de travail positif.

L'intégration : Les entreprises optent à encadrer et suivre ses employés nouvellement recrutés ou anciens par l'intermédiaire de :

- Organisation des Team Building et des sorties dans le but de forger un bon relationnel entre les personnels.
- Assurer des formations et des certifications en faveurs des collaborateurs favorisant ainsi leurs épanouissement.

1.2.1.3 Les personnels

Nous définissons les personnes chargées de maintenir les processus déjà mentionnés lors de classification des méthodes de gestion de ressources humaines. Nous citons :

- **Le directeur des ressources humaines :** c'est le responsable de la gestion des ressources humaines. Les compétences d'un directeur des ressources humaines sont

parfois intégrées dans les missions d'un directeur administratif et financier. Le directeur des ressources humaines conçoit et propose une politique d'optimisation des ressources humaines de la collectivité.

- **Les responsables RH** : Ils travaillent en collaboration avec le DRH pour mettre en oeuvre sa stratégie RH.
- **L'équipe technique** : Les ingénieurs qui développent des applications favorisant la gestion des ressources humaines et ainsi la productivité des employés.

1.2.1.4 Technologie et RH

Le Big Data et l'intelligence artificielle jouent un rôle fondamental dans la mise en oeuvre des solutions intelligentes pour le recrutement , l'analyse de productivité et la communication au sein d'une entreprise. La contribution de la technologie ne se limite pas aux logiciels permettant la standardisation, archivage et suivi des données des employés , mais aussi donner un sens à ces données et les exploiter pour en tirer des recommandations.

1.2.2 Étude de l'existant

Dans cette section, nous présenterons les solutions actuelles adoptées pour analyser et prévoir la productivité des employés par les entreprises.

1.2.2.1 IBM Watson

IBM Watson est un programme informatique d'intelligence artificielle conçu par IBM dans le but de répondre à des questions formulées en langage naturel.

IBM Watson Talent est un outil informatique qui tire parti du meilleur jugement des personnes avec les systèmes cognitif. Il permet ainsi aux RH de suivre les employés et les aider à améliorer la façon dont ils travaillent et leur impact sur l'entreprise.

On distingue des différentes solutions cognitives telles que IBM Watson Career Coach ,IBM Watson Recruitment ... qui stipulent la grande importance de l'IA et des systèmes cognitifs dans notre routine d'avenir. [4]

1.2.2.2 SP-Expert

Il s'agit d'un outil de planification flexible et éprouvée, idéal pour des secteurs dynamiques tel que la vente au détail. Le logiciel de planning permet une gestion des effectifs et une optimisation en fonctions du chiffre d'affaires et/ou nombre de visiteurs prévisionnels.

Ce logiciel est conçu par Interflex , une entreprise qui fait partie du groupe multinational Allegion. L'offre de services d'Interflex comprend principalement des solutions adaptées aux entreprises pour la gestion du personnel avec gestion du temps et planification du personnel.[5]

1.2.2.3 Critiques de l'existant

De nombreuses solutions technologiques sont présentes afin de booster la productivité au sein d'une entreprise. Cependant, une grande partie des ces solutions sont classiques et existent dans tous les départements RH. Nous remarquons aussi que les géants du domaine des TIC comme Google, IBM , Facebook se focalisent sur les applications de l'intelligence artificielle dans des différents humaine en particulier celui des ressources humaines. Mais les solutions proposées font partie des grands projets et nécessitent un effort pour pouvoir les implémenter aux entreprises et les familiariser aux responsables RH.

1.2.3 Solution proposée

Dans cette partie , nous présentons d'abord des définitions qui servent à illustrer l'utilité de l'intelligence artificielle pour le fonctionnement de notre application, ensuite nous expliquons les techniques de l'IA utilisée et finalement nous précisons la valeur ajoutée de notre solution pour les entreprises et Startup

1.2.3.1 Définitions

L'apprentissage automatique (Machine Learning) : Fait référence au développement, à l'analyse et à l'implémentation de méthodes qui permettent à une machine d'évoluer grâce à un processus d'apprentissage, et ainsi de remplir des tâches qu'il est difficile ou impossible de remplir par des moyens algorithmiques plus classiques.

Exploration de données (Data Mining) : a pour objet l'extraction d'un savoir ou d'une connaissance à partir de grandes quantités de données, par des méthodes automatiques ou semi-automatiques. Elle se propose d'utiliser un ensemble d'algorithmes issus de disciplines scientifiques diverses telles que les statistiques, l'intelligence artificielle ou l'informatique, pour construire des modèles à partir des données, c'est-à-dire trouver des structures intéressantes ou des motifs selon des critères fixés au préalable, et d'en extraire un maximum de connaissances.

1.2.3.2 Utilisation de l'IA dans notre application

La prédiction du productivité des employés utilise des algorithmes de Machine Learning. Dans notre cas , nous nous intéressons aux arbres de décision. Définition

Un arbre de décision : C'est un outil d'aide à la décision représentant un ensemble de choix sous la forme graphique d'un arbre. Les différentes décisions possibles sont situées aux extrémités des branches (les « feuilles » de l'arbre), et sont atteints en fonction de décisions prises à chaque étape. L'arbre de décision est un outil utilisé dans des domaines variés tels que la sécurité, la fouille de données, la médecine, etc. Il a l'avantage d'être lisible et rapide à exécuter. Il s'agit de plus d'une représentation calculable automatiquement par des algorithmes d'apprentissage supervisé.

Indice de GINI : Permet de mesurer le désordre (ou d'inégalité de répartition), pour choix d'un test à une position de l'arbre.

1.2.3.3 La valeur ajoutée

En tenant compte des besoins des entreprises en terme de garder une productivité significatives des personnels et en se basant sur les limites des solutions actuelles, nous proposons la conception et le développement d'un progiciel que nous baptisons PYP (Predict your productivity) qui pourra être vendus aux entreprises et aux startup pour leurs permettre de :

- Prévoir et booster la productivité des ses employés.
- Prendre les mesures nécessaires afin de garder la motivation des personnels.
- Mieux Maîtriser les données des employés et les rendre significatives.

Conclusion

Comme l'idée générale sur le cadre et le contexte du projet est désormais claire et après avoir décortiquer les intervenants ,le métier de ressource humaines et ses aspects ainsi qu'une étude de l'existant et les objectifs à atteindre, nous enchaînons avec une étude théorique concernant l'analyse des spécifications de besoins qui fera l'objet du prochain chapitre.

Analyse des besoins

Introduction

Afin de respecter les critères fixés par le cahier des charges, de garantir la réussite et l'efficacité du projet, une bonne analyse du système s'impose. Ce chapitre est consacré également à l'analyse et à la spécification des besoins fonctionnels et non fonctionnels dont nous avons besoin pour bien définir notre application. Cela va être réalisé moyennant une description détaillée et des représentations UML.

2.1 Spécification des besoins

Dans cette section, nous détaillons les acteurs de notre système ainsi que les besoins fonctionnels et non fonctionnels relatifs à notre solution.

2.1.1 Identification des acteurs

Un acteur est une entité externe qui agit sur le système. En réponse à l'action d'un acteur, le système fournit un service qui correspond à son besoin. Dans notre application nous distinguons :

- Le développeur est une personne qui a accès aux indicateurs de productivités qui illustrent la performance des employés.
- L'expert est une personne qui ajoute les indicateurs de chaque employés et consulte les prédictions de leurs productivités.

2.1.2 Les besoins fonctionnels

Un besoin fonctionnel est un besoin spécifiant une action qu'un système doit être capable d'effectuer. Par ailleurs l'ensemble de ces besoins constituent une forme de contrat envers le système cible que nous devons résoudre. Dans ce qui suit, nous présentons les fonctions que le système doit satisfaire.

Cette application permet aux développeurs de :

- Créer un compte.
- S'authentifier à travers un login et un mot de passe.
- Visualiser leurs performances ainsi que des indicateurs qui illustrent leurs productivités.
- Accéder à la base de données qui définit l'arbre de décision.

Pour les experts :

- Créer un compte.
- S'authentifier à travers un login et un mot de passe.
- L'application doit permettre la consultation des performances, en termes de productivité pour chaque employé.
- Ajouter des indicateurs qui caractérisent chaque employé.
- Consulter la prédiction en termes de performance pour chaque employé.

2.1.3 Les besoins non fonctionnels

Les besoins non fonctionnels sont les besoins qui permettent d'améliorer la qualité des services de notre solution :

- **Fiabilité** : Le temps moyen de Rétablissement de l'application est faible.
- **Ergonomie** : une présentation sous forme des interfaces plaisantes.
- **Maintenabilité** : les différents modules du système doivent être bien compréhensibles pour pouvoir les maintenir facilement et rapidement.
- **Intégrité** : L'application doit s'assurer de l'état des données afin de garantir leurs cohérence, fiabilité et pertinence.

2.1.4 Les Besoins du domaine

Les méthodes utilisées pour l'optimisation de l'agilité ne sont pas nombreuses, voir intelligentes pour prévoir et assurer une meilleure performance. En effet les entreprises utilisent des outils comme JIRA pour gérer les équipes dans la méthode agile, mais ces méthodes ne sont pas intelligentes pour présenter des solutions, prévoir et améliorer les performances. Ce qui explique la nécessité des outils intelligents pour booster la productivité des développeurs.

2.2 Modélisation

Dans cette sous section, nous allons présenter les diagrammes de cas d'utilisation de notre application. Ces diagrammes décrivent la dépendance entre le système et l'acteur afin de mieux comprendre l'aspect fonctionnel du système.

2.2.1 Langage de modélisation

Pour la modélisation objet, nous avons choisi le langage commun UML « Unified Modeling Language ». En effet, c'est un langage de modélisation et de conception des programmes informatiques. C'est un langage qui fusionne plusieurs méthodes de modélisation et de conception tel que Booch, OMT, OOSE ...

Nous avons opté pour ce langage pour faire la modélisation et la conception de notre projet parce que c'est un langage qui est standard et qui offre une grande fluidité et facilité en matière de représentation graphique des différents diagrammes.

2.2.2 Diagramme des cas d'utilisation

Les diagrammes de cas d'utilisation décrivent, sous forme d'actions et de réactions, le comportement d'un système du point de vue d'un utilisateur. Ils permettent ainsi de définir les limites du système et les relations entre le système et l'environnement.

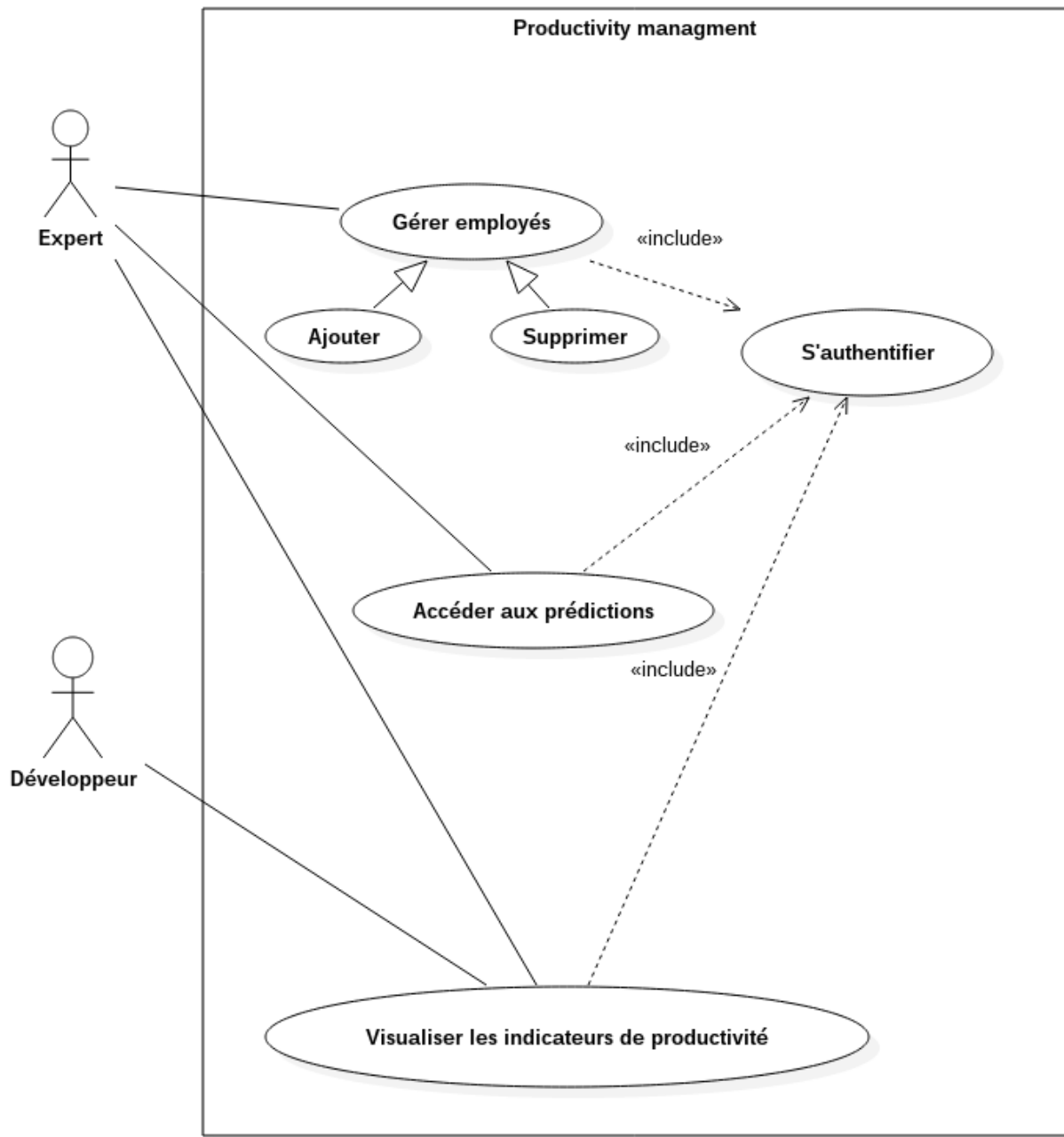


FIGURE 2.1 – Diagramme de cas d'utilisation du système

2.2.3 Diagrammes de séquences

Les principaux cas d'utilisation seront détaillés à travers quelques diagrammes de séquences afin de donner une vue dynamique sur l'interaction de l'acteur avec le système. Dans cette partie nous présentons quelques diagrammes de séquences de l'application de manière à mieux comprendre les fonctionnalités prévisibles. Au cours du chapitre suivant, nous exposerons ces diagrammes d'une façon plus détaillée afin de montrer l'interaction entre les différents modules architecturaux à concevoir.

2.2.3.1 Diagramme de séquences illustrant le scénario : "Ajouter un Employée"

La figure 2.2 présente le diagramme de séquences système du cas d'utilisation «Ajouter un Employé». Après avoir effectuer l'authentification , l'expert peut ajouter un employé en remplissant un formulaire qui contient les indicateurs de productivité de ce dernier. Le système vérifie automatiquement le format des données saisies. En cas de correspondance , la prédiction de productivité s'affiche et s'ajoute aux base de données.

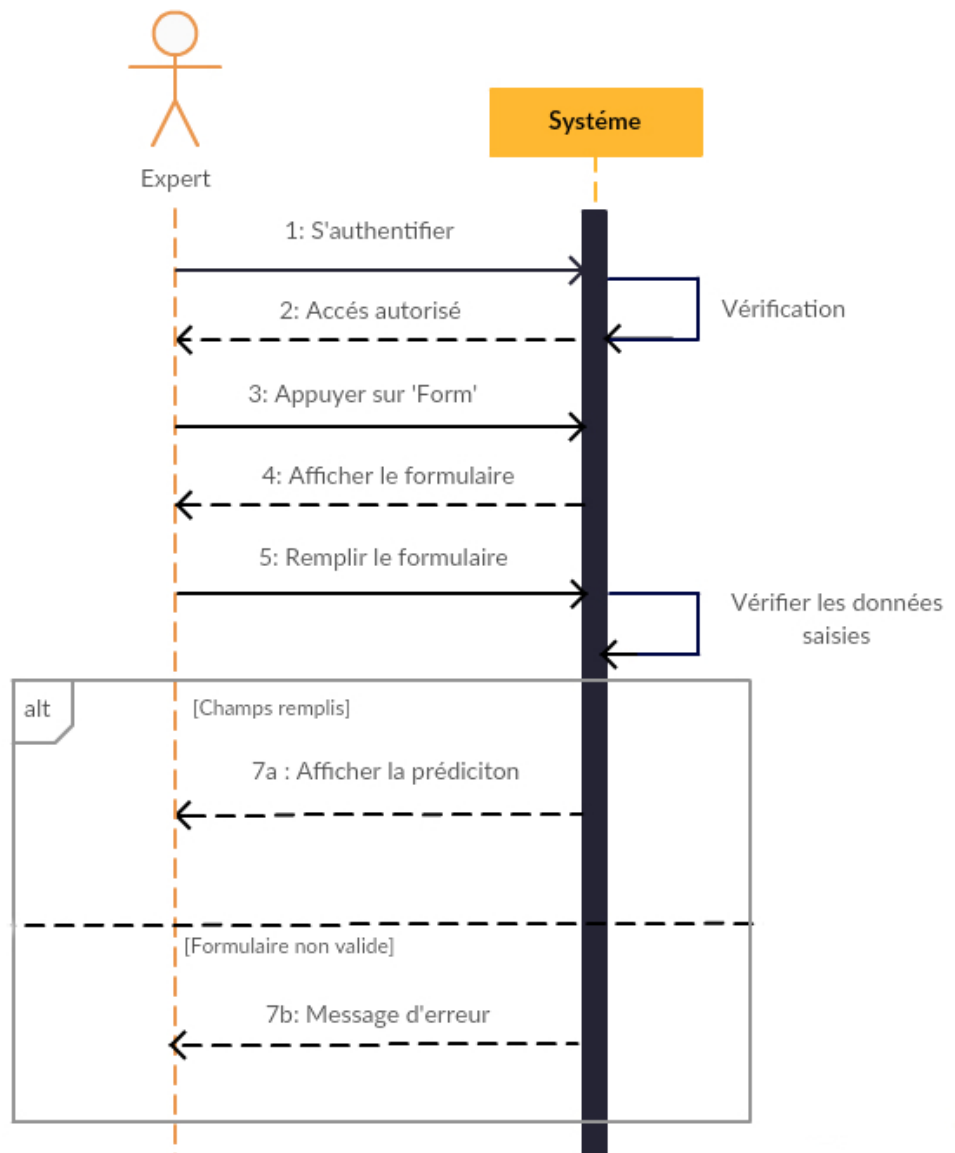


FIGURE 2.2 – diagramme de séquences système du cas d'utilisation "Ajouter un Employé"

2.2.3.2 Diagramme de séquences illustrant le scénario : "Visualiser indicateurs de performance"

La figure 2.3 décrit le diagramme de séquences système du cas d'utilisation «Visualiser indicateurs de performance».

Après avoir effectué l'authentification, l'employé ou le développeur peut visualiser des courbes qui représentent les indicateurs de performance des employés. Il suffit au développeur d'accéder à la page "chart js" et afficher les graphes d'un employé quelconque.

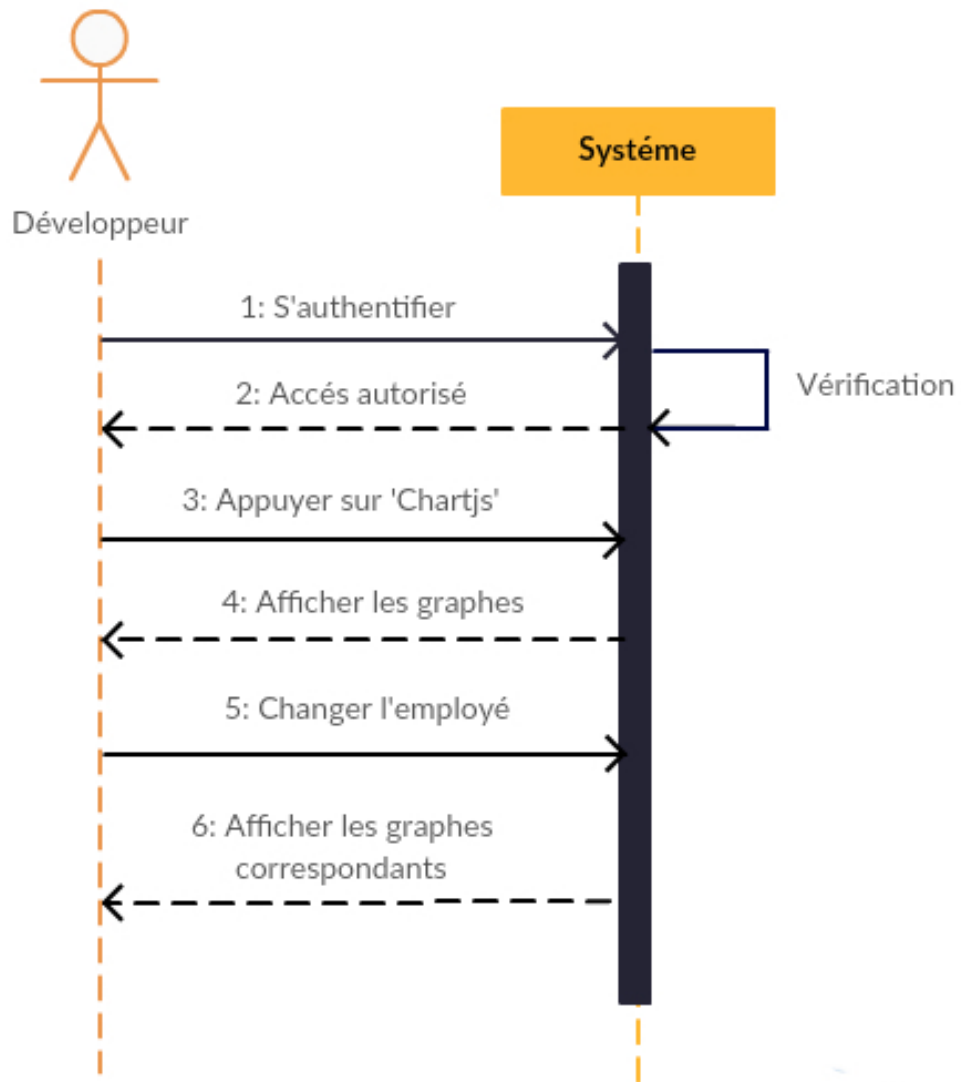


FIGURE 2.3 – diagramme de séquences système du cas d'utilisation "Visualiser indicateurs de performance"

Conclusion

Dans ce chapitre, nous avons déterminé les différents besoins que notre application est appelée à satisfaire. Puis, nous les avons illustrés par les diagrammes des cas d'utilisation

adéquats. Nous essayerons dans le chapitre qui suit de concevoir clairement l'architecture de notre système.

Conception

Introduction

Les parties d'étude de l'existant et d'analyse des besoins ont abouti à une description complète du système étudié. A cette phase, il est nécessaire de procéder à une conception et une modélisation de l'architecture générale et détaillée de l'application à réaliser. Il s'agit, donc, de la phase de conception.

3.1 Conception générale

Afin d'assurer un bon fonctionnement de l'application désirée et garantir les meilleures performances possibles, il est dans l'obligation de choisir avec le plus grand soin l'architecture globale adéquate à l'application.

Nous présentons, à travers la figure suivante, l'architecture globale de notre application :

Dans la suite, nous allons passer à la description de l'aspect physique et logique en se

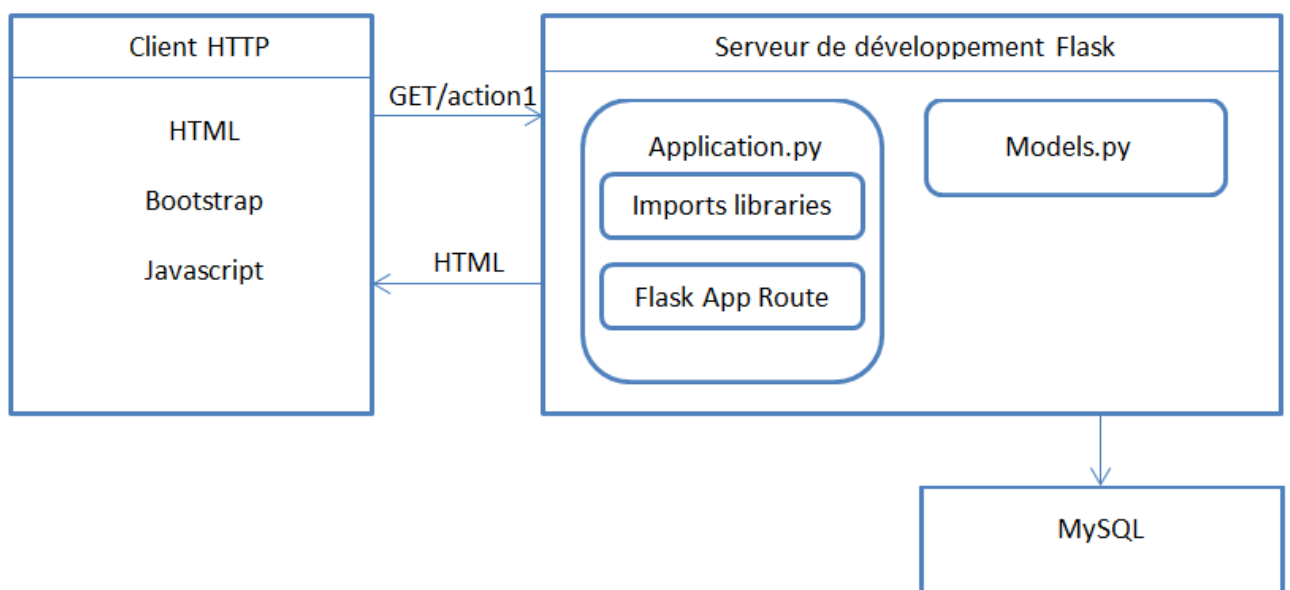


FIGURE 3.1 – Architecture globale de l'application

basant sur l'architecture physique et l'architecture logique respectivement.

3.1.1 Architecture physique

Nous avons choisi l'architecture 3-tiers comme une architecture physique pour notre application. Comme le montre la figure 3.2, l'architecture 3-tiers est composée de trois éléments, ou plus précisément dans ce cadre-là de trois couches physiques :

- La couche présentation :

Cette couche correspond à l'interface graphique avec laquelle le client (demandeur de services) interagit. Nous parlons, donc, d'Interface Homme Machine ou IHM. La couche présentation envoie les requêtes de l'utilisateur vers la couche suivante couche métier et reçoit de la part de celle-ci les résultats des traitements effectués.

- La couche métier (Le serveur d'application) :

C'est la partie fonctionnelle de l'application. Elle contient ce qu'on appelle la logique du métier qui effectue des opérations demandées par le client à travers la couche présentation. La couche métier offre alors des services applicatifs et métier à la couche présentation. Pour fournir ces services, elle s'appuie, le cas échéant, sur les données du système, accessibles à travers des services de la couche inférieure. En retour, elle renvoie à la couche présentation les résultats qu'elle a calculés.

- La couche accès aux données (Le serveur de base de données) :

C'est la partie du système qui gère l'accès aux données stockés.

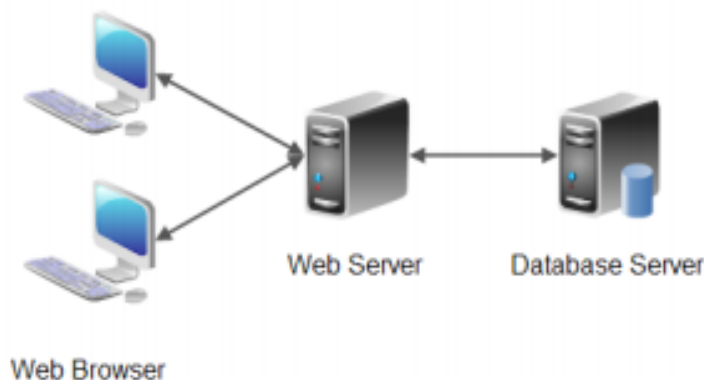


FIGURE 3.2 – Architecture trois-tiers

3.1.2 Architecture logique

Nous avons choisi comme architecture logicielle pour notre application l'architecture MVT [6] qui est une architecture orientée autour de trois pôles : le modèle, la vue et le template. Elle s'inspire de l'architecture MVC, très répandue dans les frameworks web. Son objectif est de séparer les responsabilités de chaque pôle afin que chacun se concentre sur ses tâches.

- Le modèle :

Le modèle interagit avec la base de données. Sa mission est de chercher dans une base de données les items correspondant à une requête et de renvoyer une réponse facilement exploitable par le programme. Les modèles s'appuient sur un ORM (Object Relational Mapping, ou Mapping objet-relationnel).

Dans notre application, les modèles vont gérer toutes les données qui se trouvent dans une base de données.

- La vue :

La vue fait l'interface avec l'utilisateur. Sa première tâche est d'afficher les données qu'elle a récupérées auprès du modèle. Sa seconde tâche est de recevoir toutes les actions de l'utilisateur (clic de souris, sélection d'une entrée, ...). Ces différents événements sont envoyés au template.

Dans notre application, les vues constituent les pages Web de notre site. La vue n'effectue aucun traitement.

- Le template :

Un template est un fichier HTML qui peut recevoir des objets Python et qui est lié à une vue. Concrètement, un template peut interpréter des variables et les afficher. Il sera récupéré par la vue et envoyé au visiteur. Cependant, avant d'être envoyé, il sera analysé et exécuté par le framework.

Nous illustrons le modèle par le schéma d'exécution d'une requête suivant.

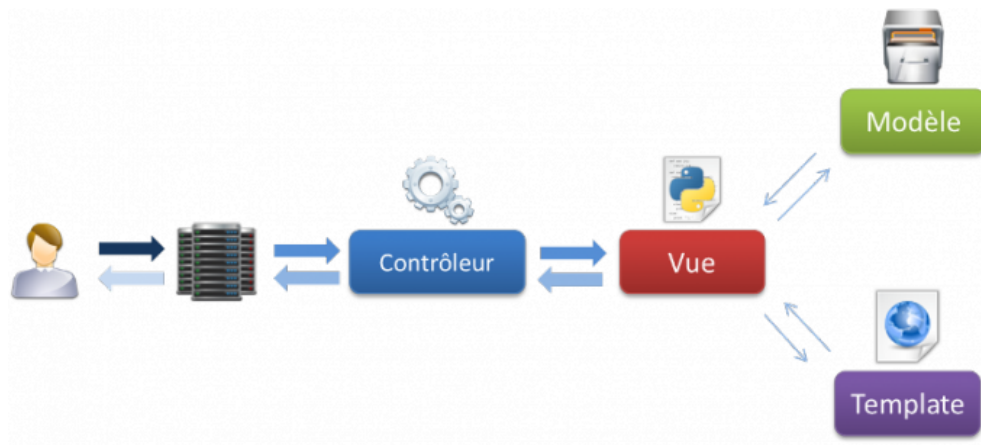


FIGURE 3.3 – Patron de conception MVT

3.1.3 Choix de l'architecture

Actuellement, l'utilisation d'une architecture 3-tiers en parallèle avec une architecture MVT pour le développement des applications web est devenue courante spécialement lorsqu'il s'agit d'une application basée sur des technologies python. Dans ce cas l'architecture 3-tiers est considérée comme une architecture physique alors que l'architecture MVT est considérée comme une architecture logique. Une telle configuration assimile d'une part le modèle de MVT à la couche accès aux données de l'architecture 3-tiers et lui alloue un serveur de base de données, d'autre part le contrôleur est considéré comme la couche métier et enfin la vue est comparée à la couche présentation.

Une architecture pareille offre comme avantages :

- Une conception claire et efficace grâce à la séparation des données.
- Responsabilité unique, couplage faible et cohésion forte
- Un gain de temps de maintenance et d'évolution de l'application.
- Une plus grande souplesse pour organiser le développement de l'application entre différents développeurs
- Amélioration du niveau de sécurité (La sécurité peut être définie pour chaque service)
- Une meilleure testabilité (les fonctionnalités peuvent être testées indépendamment de l'interface)

Il est, désormais, dans l'évidence que les architectures modernes reposent sur les modèles distribués. Celles-ci favorisent une gestion meilleure et une maintenance plus simple. D'autre part, les systèmes modernes sont de plus en plus interactifs, nécessitant alors une architecture qui favorise la gestion des événements et l'interaction. Ce qui justifie, amplement, notre choix en termes d'architecture de l'application à développer.

3.1.4 Le diagramme de paquetage

Un paquetage est un conteneur logique permettant de regrouper et d'organiser les éléments dans le modèle UML. Le diagramme de paquetage sert ainsi à représenter les dépendances entre les ensembles de définitions. Cette représentation nous permettra d'améliorer les facteurs de couplage et cohésion de notre système c'est-à-dire la forte dépendance intra-paquetages et le faible échange inter-paquetages. La figure 3.4 illustre la représentation à haut niveau de l'organisation de notre application.

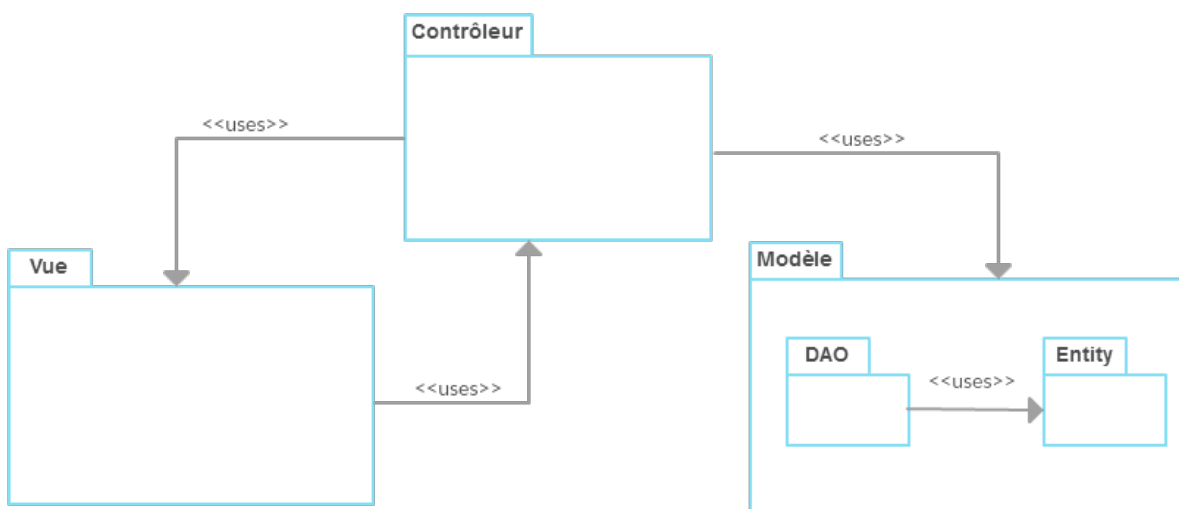


FIGURE 3.4 – Diagramme de paquetage

3.1.4.1 Description des paquetages

Pour l'organisation de notre application, nous avons intégré trois paquets :

- **Le paquetage « Modèle »** : Les fichiers du paquetage Modèle contiennent les structures des entités de la base de données.

- **Le paquetage « Contrôleur »** : Ce paquet contient la logique métier de l'application. Ses fichiers servent comme un lien entre les modèles et les vues.

- Le **paquetage « Vue »** : Contient les pages qui doivent être retournées par l'application après action du contrôleur.

3.2 Conception détaillée

Une fois l'architecture globale est définie, il est maintenant nécessaire de concevoir le modèle des données adéquat à l'application. Ce modèle convient à retrouver les différents types d'objets recensés dans cette application et de les modéliser selon un formalisme bien précis. Il s'agit donc de concevoir la base de données et modéliser l'aspect dynamique du système en dressant le diagramme de séquence des cas d'utilisation les plus importants.

3.2.1 Conception de la base de données

3.2.1.1 Modèle Entité-Association

L'élaboration d'un modèle Entité-Association est une étape incontournable dans le processus de création de la base de données. Notre application nécessite une base de données relationnelle qui permet de stocker toutes les données du système étudié.

La figure suivante représente le diagramme Entité-Association de notre application.

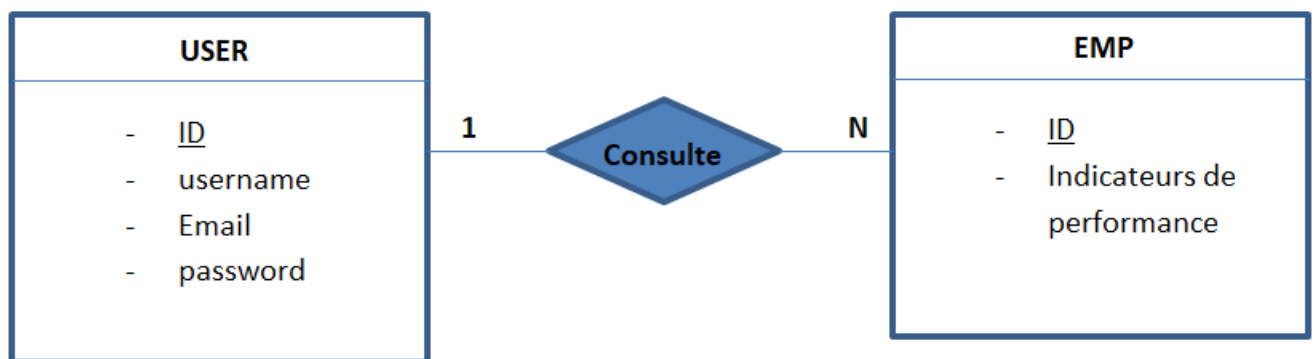


FIGURE 3.5 – Diagramme Entité-Association

3.2.1.2 Identification des entités

Les entités renferment des attributs et des clés primaires qui seront représentés dans le tableau suivant. Il est à noter que les attributs de la table EMP, autre que la clé primaire, ont été mentionnés sous l'expression d' « indicateurs de performance » afin de garantir la clarté de la figure.

| Entité | Identifiant | Attributs | Libellé |
|--------|-------------|---------------------------|--|
| User | ID | username | Nom d'utilisateur |
| | | Email | L'adresse mail de l'utilisateur |
| | | password | Le mot de passe de l'utilisateur |
| EMP | ID | Age | L'âge de l'employé |
| | | Attrition | Taux d'attrition de l'employé |
| | | BusinessTravel | Les missions de travail à l'étranger |
| | | DailyRate | Le taux quotidien de l'employé |
| | | Department | Le département de l'employé |
| | | Distance from Home | La distance séparant le lieu d'habitat de l'employé de son lieu de travail |
| | | Education | Le niveau d'éducation de l'employé |
| | | EducationField | Le domaine d'éducation de l'employé |
| | | EmployeeCount | Composition de l'effectif |
| | | EmployeeNumber | Le numéro de l'employé |
| | | EnvironnementSatisfaction | La satisfaction de l'employé vis-à-vis son environnement |
| | | Gender | Le sexe de l'employé |
| | | HourlyRate | Le taux horaire de l'employé |

TABLE 3.1 – Tableau d'identification des entités

3.2.1.3 Identification des associations

Une seule association a été perçue dans le système étudié. Nous la présentons dans le tableau suivant :

| Nom de l'association | Entités impliquées | Type | Description |
|----------------------|--------------------|-------|---|
| Consulter | USER- EMP | 1 : N | Un utilisateur peut consulter les profils de plusieurs employés |

TABLE 3.2 – Tableau d'identification des associations

3.2.1.4 Modèle logique des données

Le Modèle logique des données consiste à décrire la structure de données utilisée sans faire référence à un langage de programmation. C'est un modèle relationnel créé à partir d'un modèle entité-association. Le modèle logique décrivant notre base de données est le suivant (**les clés primaires sont soulignées et les clés étrangères sont précédées par '#'**) :

USER(ID, username, Email, password)

EMP(#ID, Age, Attrition, BusinessTravel, DailyRate, Department, Distance from Home, Education, EducationField, EmployeeCount, EmployeeNumber, EnvironementSatisfaction, Gender, HourlyRate)

3.2.2 Diagrammes de séquences d'objets

Pour modéliser l'aspect dynamique du système, nous présentons dans ce paragraphe quelques diagrammes de séquences. En effet, un diagramme de séquences a pour but de décrire les relations entre objets d'un point de vue temporel en mettant l'accent sur la chronologie des envois de messages. Nous présentons dans ce qui suit les diagrammes de séquences «Ajouter un employé» et «Visualiser les indicateurs de performance» représentés respectivement par les figures 3.6 et 3.7.

3.2.2.1 Diagramme de séquences objets «Ajouter un employé»

La figure 3.6 ci-dessous résume les différentes étapes par lesquelles passe le système afin d'ajouter un employé à la base de données. En effet, la première étape est celle de

l'enregistrement de l'utilisateur. Lorsque nous appuyons sur le bouton « Login », la requête http passe du navigateur web au serveur de développement Flask. L'authentification passe par une vérification des données à travers MySQL Server. Une fois nous avons l'accès à la page d'accueil, nous pourrions ouvrir le formulaire. Les mêmes étapes de l'envoi de la requête HTTP et de sa redirection auront lieu. L'expert pourrait donc remplir le formulaire d'adhésion.

Au final, les informations envoyées à travers le formulaire seront traitées afin de déterminer la prédiction de l'employé qui s'afficherait à la page d'accueil en cas de réussite, sinon un message d'erreur serait affiché.

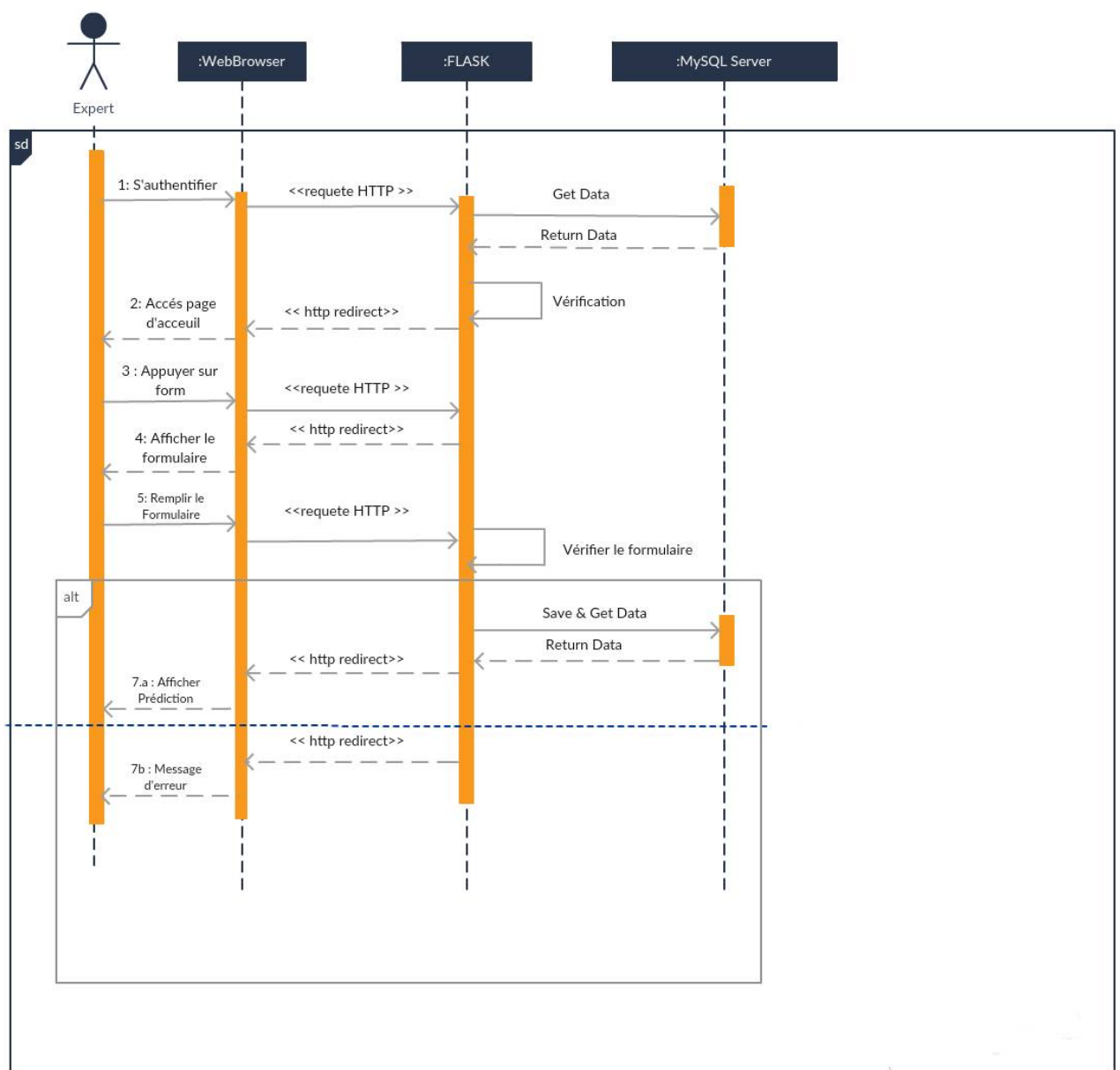


FIGURE 3.6 – Diagramme de séquences objets « Ajouter un employé »

3.2.2.2 Diagramme de séquences objets « Visualiser indicateurs de performance »

Après avoir effectué l'authentification, le développeur est redigé directement à la page d'accueil. En cliquant sur le lien "chartjs", une requete HTTP sera envoyée automatiquement au serveur de développement FLASK, qui lui meme retourne la route vers la page désirée. le développeur consultera ainsi les graphes qui traduisent les indicateurs de productivité.

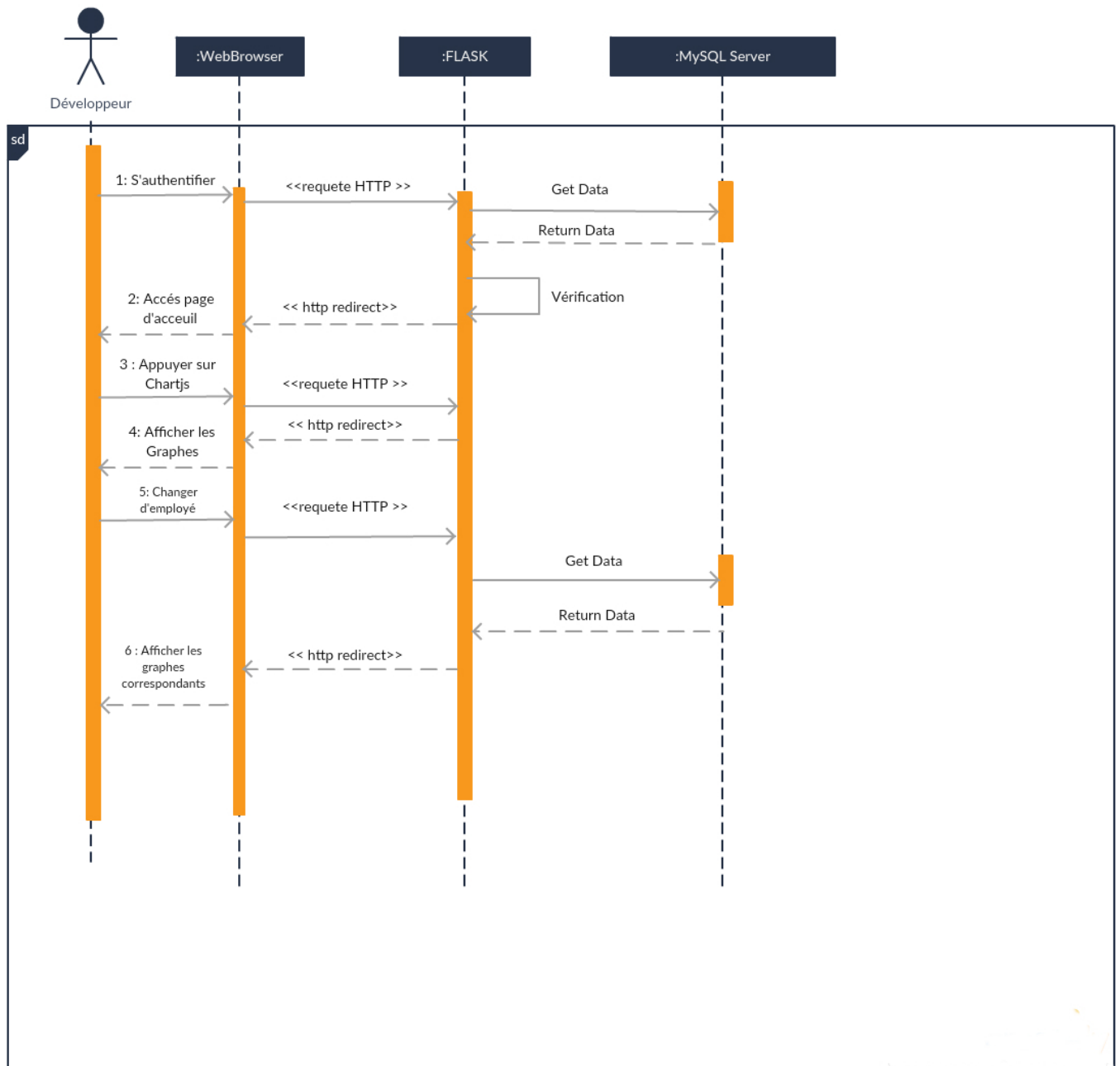


FIGURE 3.7 – Diagramme de séquences objets « Visualiser indicateurs de performance »

Conclusion

Dans ce chapitre, nous avons commencé par donner une conception architecturale de notre système. Ensuite, nous avons présenté une conception détaillée à l'aide des diagrammes UML. Cette phase est nécessaire pour faciliter l'étape de la réalisation que nous allons la présenter dans le chapitre suivant.

Introduction

Après avoir achevé l'étape de la conception, en tenant compte des besoins fixés et des choix conceptuels effectués, nous consacrons ce chapitre à la description du travail réalisé. Nous commençons par décrire l'environnement matériel et logiciel sur lequel nous avons développé notre application. Ensuite, nous signalons l'apport de l'intelligence artificielle sur le déroulement de notre application, après nous identifions l'état d'avancement du projet et nous mettons en évidence le travail réalisé par la présentation de quelques captures d'écrans traduisant le déroulement du projet. Enfin, nous finissons par un chronogramme qui décrit toutes les étapes de mise en oeuvre du travail.

4.1 Environnement de travail

Dans cette partie nous nous intéressons à l'étude de l'environnement technique disponible pour la réalisation de notre projet, ensuite nous justifions les choix pris en matière d'environnement logiciel pour mener à terme la partie applicative.

4.1.1 Environnement matériel

Le développement de l'application est réalisé sur deux ordinateurs ayant les caractéristiques suivantes :

— **Première station de travail :**

- Marque : Laptop Lenovo Z50-70
- Processeur : Intel(R) Core(TM) i5-4210U CPU @ 1.70GHz 2.40 GHz
- Mémoire : 6,00 Go de RAM
- Disque Dur : 1 TB
- Ecran : 17

— **Deuxième station de travail :**

- Marque : ASUS
- Processeur : Intel Core i5 CPU
- Mémoire : 8,00 Go de RAM
- Disque Dur : 1 TB
- Ecran : 15,6

4.1.2 Environnement logiciel

Cette section présente les différents outils logiciels que nous avons exploité pour réaliser notre application. Pour la réalisation de notre projet, nous avons opté pour l'utilisation de l'environnement logiciel suivant :

- **Sublime Text** : Sublime Text est un éditeur de texte générique codé en C++ et Python, disponible sur Windows, Mac et Linux. Le logiciel a été conçu tout d'abord comme une extension pour Vim, riche en fonctionnalités. Sublime Text intègre la plupart des fonctionnalités de base d'un éditeur de texte, dont la coloration syntaxique personnalisable, l'auto complétion, un système de plugins... L'éditeur propose cependant des fonctions plus avancées, dont :
 - Minimap : prévisualisation de tout le fichier dans une barre latérale.
 - Sélection et édition dans plusieurs sections de code en parallèle.
 - Marque-page au sein même des fichiers.
 - Sauvegarde automatique.
 - Recherche et remplacement par expressions régulières.
 - Support des macros et de plugins en Python.
- **MySQL** : MySQL est un Système de Gestion de Bases de Données Relationnelles (abrégié SGBD). C'est-à-dire un logiciel qui permet de gérer des bases de données, et donc de gérer de grosses quantités d'informations. Il utilise pour cela le langage SQL. Il s'agit d'un des SGBDR les plus connus et les plus utilisés. MySQL peut donc s'utiliser seul, mais est la plupart du temps combiné à un autre langage de programmation : PHP par exemple pour de nombreux sites web, mais aussi Java, Python, C++, et beaucoup, beaucoup d'autres.

- **ShareLatex** : Environnement de développement LATEX utilisé pour rédiger ce rapport. Il permet de créer, éditer, partager un document latex et ce en ligne, sans aucune installation préalable.
- **Creately** : C'est un outil de modélisation UML, gratuit sur le plan public. Il contient une interface épurée combinée avec des fonctionnalités uniques, il aide à dessiner des diagrammes jusqu'à 3 fois plus vite par rapport à un logiciel de diagramme classique.

4.1.3 Choix technologiques

Dans ce paragraphe nous présentons les différentes technologies adaptées pour la réalisation de notre application.

- **Python** : C'est un langage de programmation objet, multi-paradigme et 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 est ainsi similaire à Perl, Ruby, Scheme, Smalltalk et Tcl. Le langage Python est placé sous une licence libre proche de la licence BSD 4 et fonctionne sur la plupart des plateformes informatiques, des supercalculateurs aux ordinateurs centraux⁵, de Windows à Unix avec notamment GNU/Linux en passant par macOS, ou encore Android, iOS, et aussi avec Java ou encore .NET. Il est conçu pour optimiser la productivité des programmeurs en offrant des outils de haut niveau et une syntaxe simple à utiliser. Il est également apprécié par certains pédagogues qui y trouvent un langage où la syntaxe, clairement séparée des mécanismes de bas niveau, permet une initiation aisée aux concepts de base de la programmation.
- **Flask** : Flask est un framework open-source de développement web en Python. Son but principal est d'être léger, afin de garder la souplesse de la programmation Python, associé à un système de templates. Il est distribué sous licence BSD. Parmi ces fonctionnalités :
 - Contient un serveur de développement et un debugger.
 - Supporte les tests unitaires.
 - Marque-page au sein même des fichiers.
 - Utilise le moteur de template Jinja2.
 - Supporte les cookies sécurisés (session).

- **SQLAlchemy** : C'est un toolkit open source SQL et un mapping objet-relationnel (ORM) écrit en Python et publié sous licence MIT.

4.2 Explication du travail réalisé

Dans cette partie , nous détaillerons l'aspect intelligence artificielle utilisé dans notre application web.

4.2.1 Cas de test de prédiction

En se référant au chapitre 3 , nous avons réutilisé la base de données d'IBM [7] pour la définition de notre modèle.

La spécification du modèle est défini comme suit :

- L'apprentissage : Nous utiliserons 85%. des données pour le "Training" de l'arbre de décision.
- Le test : 15% seront utilisées pour les cas de test. Nous déduisons ainsi la précision en terme de prédiction de performance du modèle défini.

La définition du modèle dans notre système utilise les indicateurs ou features suivants :

"Age", "DailyRate" , "DistanceFromHome" , "Education", "EmployeeNumber", "StandardHours" , "TotalWorkingYears" , "TrainingTimesLastYear" , "WorkLifeBalance" , "YearsAtCompany" , "YearsSinceLastPromotion" , "YearsWithCurrManager" , "EnvironmentSatisfaction" et notre cible ou Target est : "PerformanceRating"

En se basant sur ce modèle , nous aurons l'arbre de décision suivant :

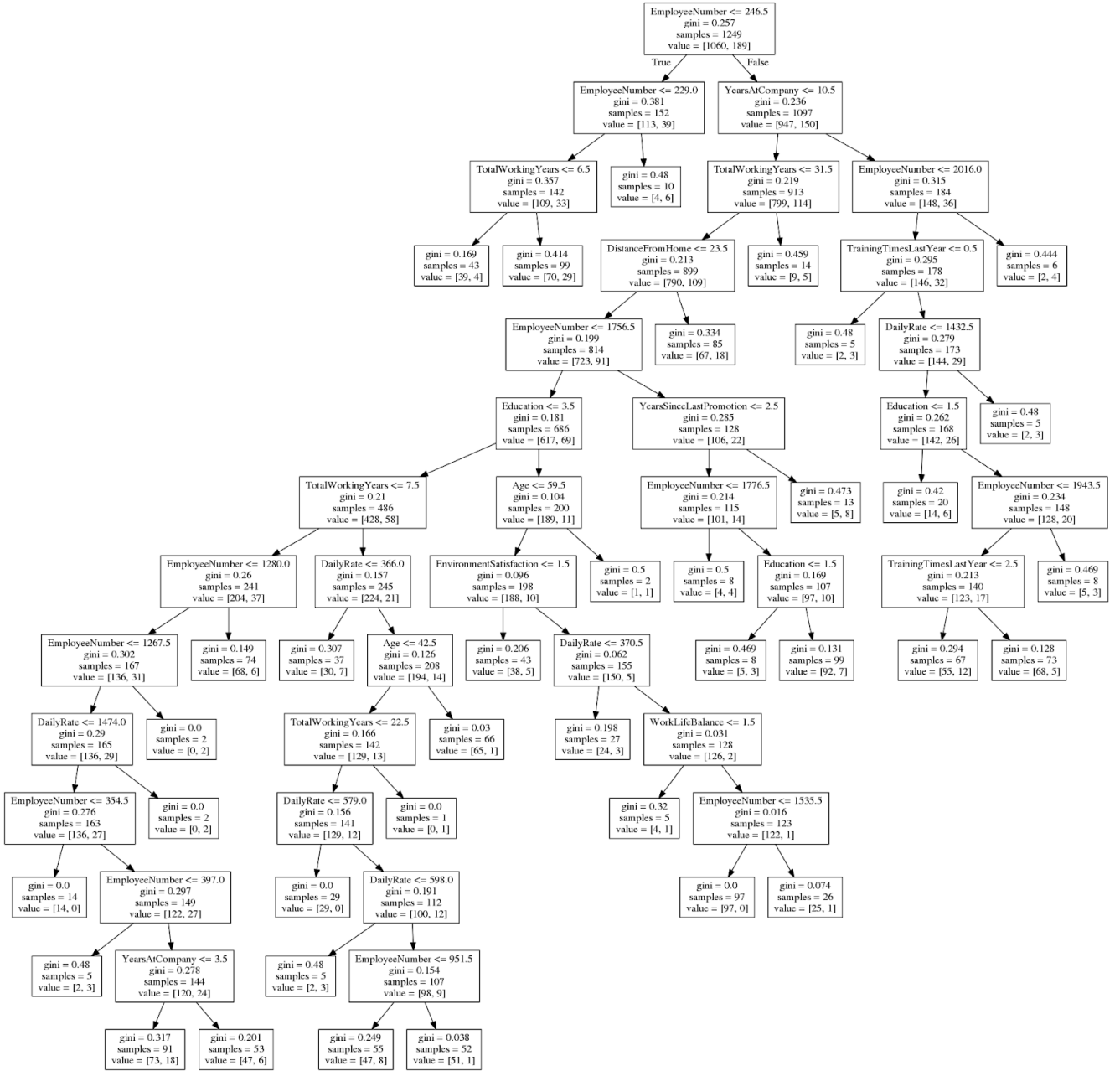


FIGURE 4.1 – Arbre de décision

Nous utiliserons maintenant comme indicateurs de performances les valeurs suivantes :

- Age : 40
- DailyRate : 1100
- DistanceFromHome : 3

- Education : 5
- EmployeeNumber : 600
- StandardHours : 80
- TotalWorkingYears : 8
- TrainingTimesLastYear : 2
- WorkLifeBalance : 3
- YearsAtCompany : 4
- YearsSinceLastPromotion : 2
- YearsWithCurrManager : 2
- EnvironmentSatisfaction : 4

Le déroulement de l'algorithme d'arbre de décision nous donne cette figure : Chaque noeud correspond à une décision à prendre par le système.

Dans notre cas : $\text{EmployeeNumber} = 600 > 246.5$ donc la décision = FALSE . $\text{YearsAtCompany} = 4 \leq 10.5$ donc la décision = TRUE Nous continuons de la même manière jusqu'à atteindre une feuille. Si dans ce cas $\text{gini}=0.0$, nous déduisons que notre modèle a pu distinguer entre les deux classes , qui sont dans notre cas la classe qui illustre une $\text{performance}=3$ et la classe qui illustre une $\text{performance}=4$.

Nous remarquons d'après l'arbre de décision que $\text{gini}=0.0$, et $\text{values} = [97,0]$, ainsi la productivité de l'employé est 3 .

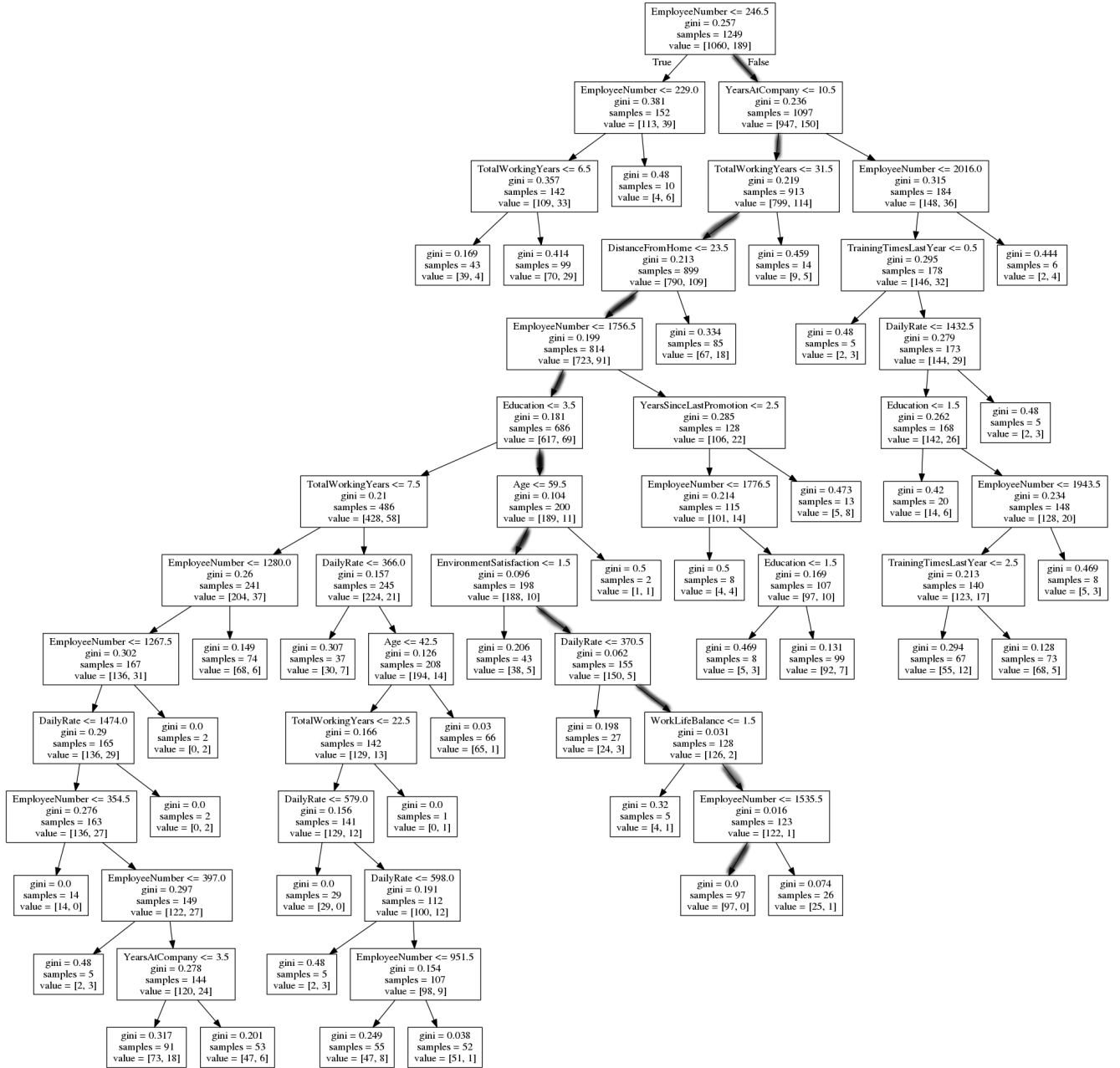


FIGURE 4.2 – Parcours de l'Arbre de décision

4.3 Aperçu sur le travail réalisé

Dans cette section, nous exposons l'application réalisée en montrant ses différentes interfaces graphiques et les fonctionnalités offertes pour l'utilisateur.

4.3.1 Page d'authentification

C'est la première page de l'application , elle constitue aussi l'interface commune à tous les utilisateurs qui leur permet de s'authentifier en saisissant chacun son nom d'utilisateur et son mot de passe comme le montre la figure. .

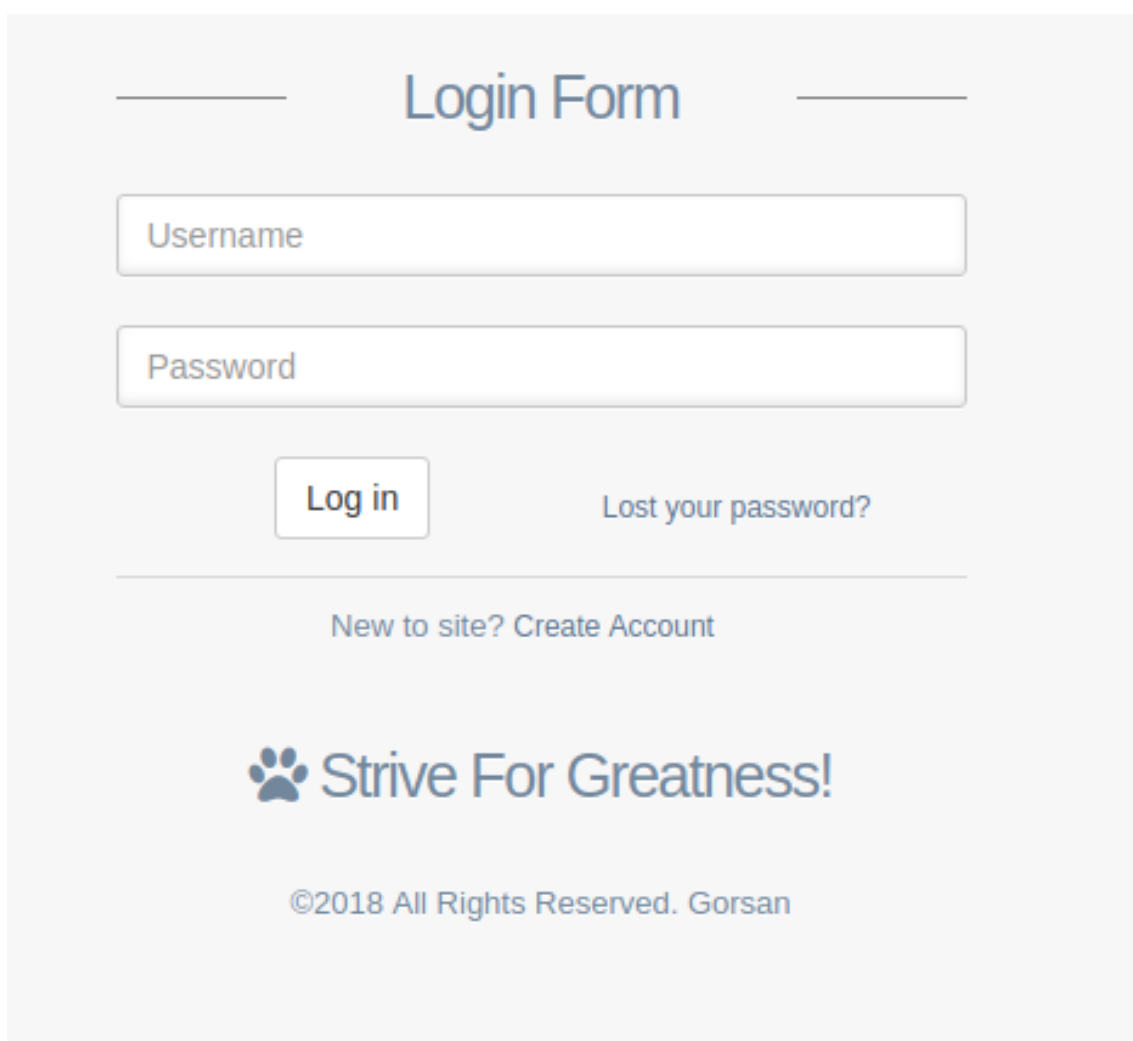
The image shows a web interface for a login form. At the top, the title "Login Form" is centered in a blue font, flanked by two horizontal lines. Below the title are two input fields: the first is labeled "Username" and the second is labeled "Password". Both fields have a light gray border and a subtle shadow. Below the "Password" field is a "Log in" button with a blue border and text. To the right of the button is a link "Lost your password?" in blue text. Below these elements is a horizontal line, followed by the text "New to site? Create Account" in blue. At the bottom, there is a paw print icon followed by the text "Strive For Greatness!" in a large blue font. Below that is the copyright notice "©2018 All Rights Reserved. Gorsan" in a smaller blue font.

FIGURE 4.3 – Interface de login

Si les paramètres d'authentification sont non valides, un message d'erreur sera affiché. Dans le cas contraire, l'utilisateur sera dirigé vers la page d'accueil.

4.3.2 Page d'accueil

Une fois connecté, une session s'ouvre et l'utilisateur se trouve face à une page d'accueil bien claire et bien organisée facilitant toutes sortes d'interaction. La page d'accueil représentée par la figure 4.4 contient principalement la barre de menu des différentes fonctionnalités de l'application à savoir l'ajout d'un employé, la visualisation de la table utilisée pour définir l'arbre de décision et la tables des employés ajoutés, et des graphes illustrant les indicateurs de performances.

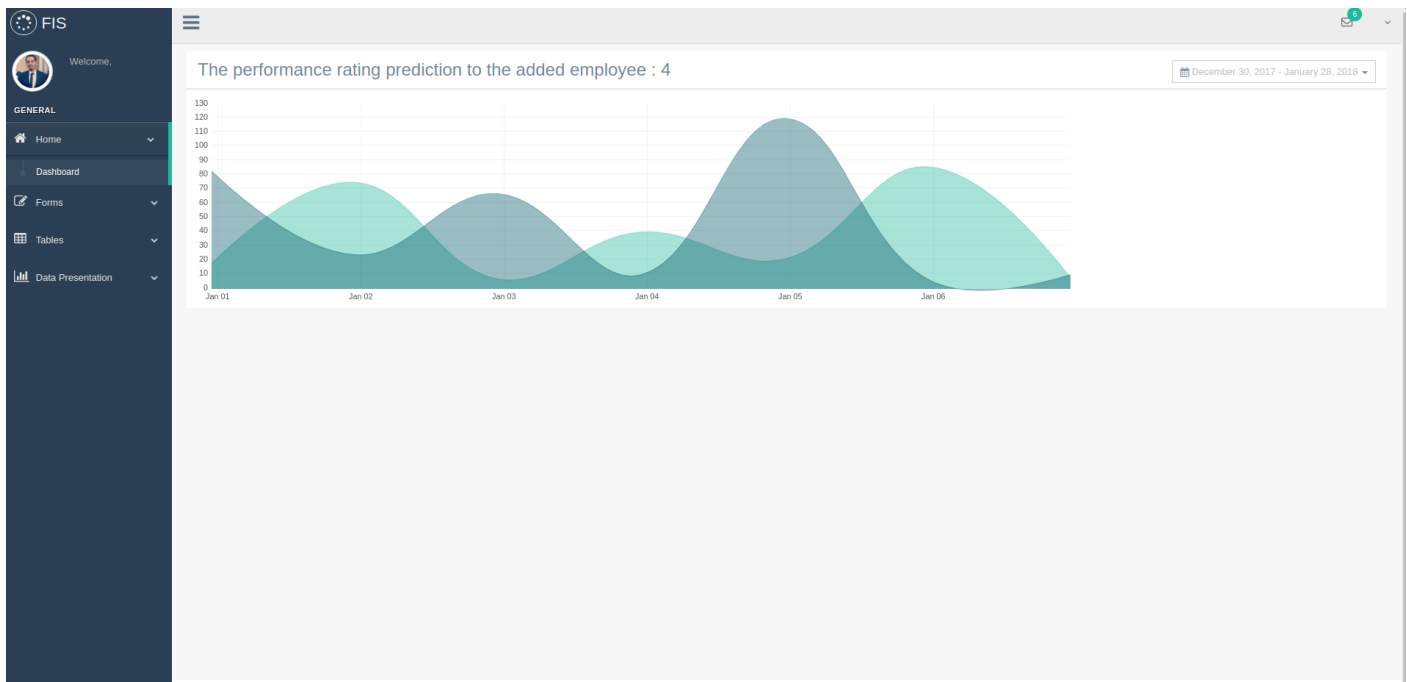


FIGURE 4.4 – Interface d'accueil

4.3.3 Interface d'ajout des employés par un expert

Après avoir authentifié, l'expert peut consulter la page form en cliquant sur "General Form". Il pourra donc ajouter un employé en remplissant ses critères de productivités.

FIGURE 4.5 – Interface d’ajout des employés par un expert

4.3.4 Interface d’affichage des employés

Après l’ajout d’un employé , l’utilisateur peut visualiser la tables des employés contenant leurs indicateurs de performances. Il pourra aussi consulter la table utilisée pour construire l’arbre de décision.

| # | First Name | Last Name | Username |
|---|------------|-----------|----------|
| 1 | Mark | Otto | @mdo |
| 2 | Jacob | Thornton | @fat |
| 3 | Larry | the Bird | @twitter |

| ID | Age | PerformanceRating | JOBsatisfaction | DistanceFromHome | Salary | Relationship |
|-----------|-----|-------------------|-----------------|------------------|----------|--------------|
| 121000040 | 50 | 4 ↑ | 5 | 3 | \$7.45 | single |
| 121000039 | 45 | 4 ↑ | 2 | 2 | \$741.20 | single |
| 121000038 | 30 | 3 ↑ | 6 | 4 | \$432.26 | married |
| 121000037 | 25 | 4 | 8 | 4 | \$333.21 | single |
| 121000040 | 5 | 4 | 7 | 2 | \$7.45 | single |
| 121000039 | 4 | 3 ↓ | 9 | 4 | \$741.20 | married |

FIGURE 4.6 – Interface d’affichage des indicateurs des employés

4.3.5 Interface d’affichage des graphes

L’employé peut visualiser des graphes qui illustrent les indicateurs de sa performance. Il suffit de cliquer sur “Data Presentation” puis sur “chartjs” et sera redirigé vers cette interface.

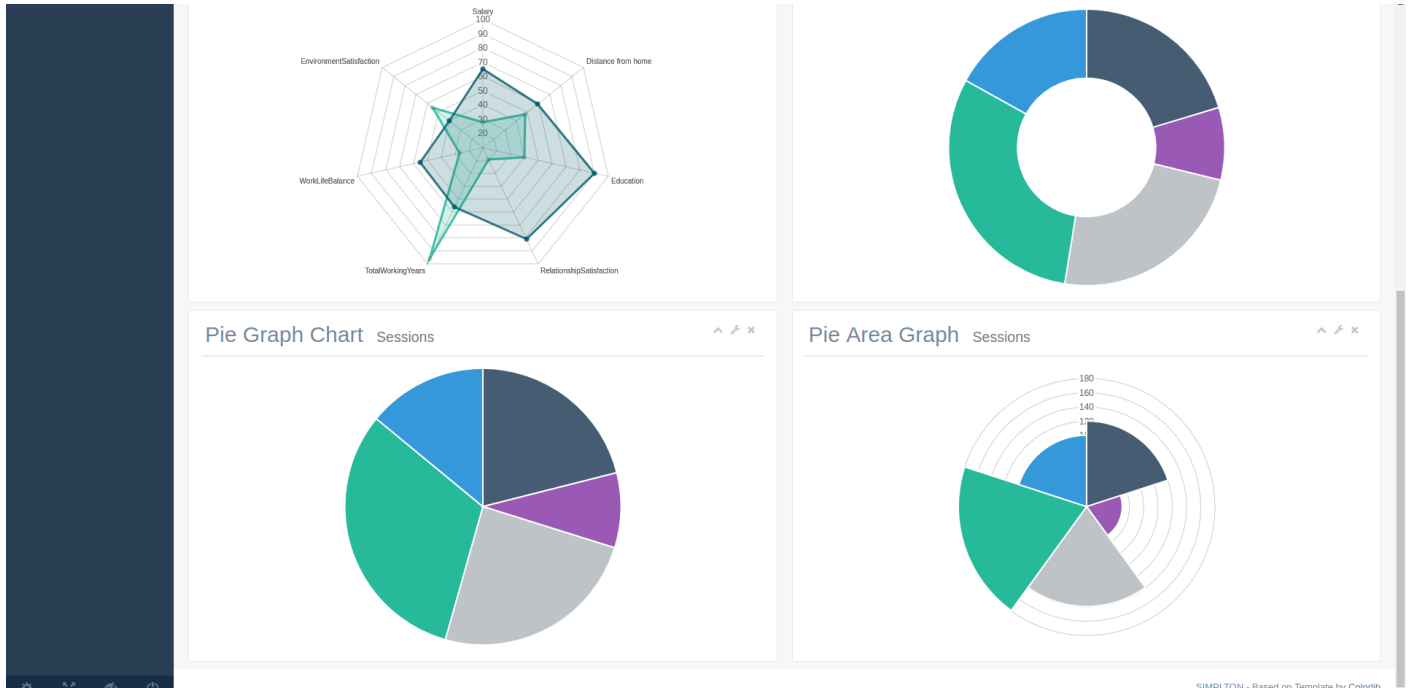


FIGURE 4.7 – Interface graphique illustrant les critères de performance

4.4 Chronogramme de gestion de travail

Ce projet a été réalisé pendant une durée de 4 mois. Nous avons tout d'abord commencé par une étude approfondie de l'existant pour pouvoir comprendre les besoins et débiter le travail. Ensuite, après avoir dégagé et compris ce qui est demandé, nous avons pu relever les spécifications fonctionnelles et non fonctionnelles. Puis nous avons entamé la phase de conception et implémenté le module demandé après avoir passé par une phase de documentation sur les technologies à utiliser. Dans un souci d'organisation et afin de mener à bien le projet, nous avons établi un chronogramme détaillé des différentes tâches qui constituaient le projet.

| Mois | Février | | | | Mars | | | | Avril | | | | Mai | | | |
|---|---------|---|---|---|------|---|---|---|-------|---|---|---|-----|---|---|---|
| Semaine | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
| Spécification et réalisation de cahier des charges | | | | | | | | | | | | | | | | |
| Documentation et familiarisation avec les outils de développement | | | | | | | | | | | | | | | | |
| Conception | | | | | | | | | | | | | | | | |
| Implémentation | | | | | | | | | | | | | | | | |
| Rédaction du rapport | | | | | | | | | | | | | | | | |

FIGURE 4.8 – Chronogramme de gestion de travail

Conclusion

Dans ce chapitre de la réalisation, nous avons présenté les environnements matériels et logiciels sur lesquels nous avons développé notre projet, ainsi que les technologies employées. Nous avons, par la suite, présenté l'apport de l'intelligence artificielle sur notre travail et une illustration d'application de l'algorithme Arbre de décision pour prévoir la performance d'un employé. Finalement, nous avons présenté les interfaces les plus significatives de notre application en clôturant par le chronogramme des tâches accomplies durant ce projet.



Conclusion générale et perspectives

Le travail présenté dans le présent rapport, a été réalisé dans le cadre du projet de conception et de développement pour les élèves ingénieurs de la deuxième année de l'ENSI. Nous avons détaillé, tout au long des quatre chapitres, les différentes étapes qui ont menées à la concrétisation du projet. En effet, dans le premier chapitre, nous avons commencé par présenter l'organisme d'accueil ainsi que le travail demandé. Ensuite, Nous nous sommes focalisés sur l'état de l'art en passant par une étude de l'existant et une présentation du contexte du travail à réaliser et finalement nous avons présenté notre solution. Plus tard, nous avons dédiés le deuxième chapitre à la spécification des besoins en illustrant par des diagrammes des cas d'utilisation. Cette étape avait pour objectif de fixer les fonctionnalités que notre application est destinée offrir aux utilisateurs. Dans le troisième chapitre, nous avons déterminé la conception de l'application selon l'architecture adoptée. Tout au long de ce projet, nous sommes arrivés à implémenter plusieurs fonctionnalités. La première fonctionnalité consiste à gérer les employés en ajoutant,modifiant et supprimants leurs critères de performance. La deuxième focntionnalité , qui est le noyau de notre application permet de prévoir la productivité de chaque employé. la troisième fonctionnalité permet aux développeurs de suivre leurs indicateurs de performance grâce aux représentations graphiques. Ce projet s'est avéré opportun sur plusieurs volets. En fait, au cours de la réalisation de ce travail, nous avons eu l'occasion de visiter à plusieurs reprises l'une des entreprises les plus réputées dans le domaine des TIC pas seulement en Tunisie, mais aussi à travers le monde. Ce qui fait que nous avons découvert la culture d'entreprise ainsi que leurs besoins en termes de gestion interne de leurs ressources humaines. Cette expérience fut très enrichissante sur le plan personnel et professionnel à la fois. Travailler sur des technologies bien demandées sur le marché professionnel était une occasion pour acquérir des connaissances solides qui ne peuvent que fortifier la valeur ajoutée tirée par une telle expérience. Notre application ne fait pas l'exception en termes d'amélioration et d'optimisation durant la période à venir. Plusieurs opportunités sont à notre disposition

afin d’aller plus loin dans son développement. Ce que nous pouvons proposer, à l’instant présent, est une fonctionnalité de plus qui pourrait améliorer le service offert. En fait, la détection des relations entre les différents critères agissant sur la performance des développeurs pourrait bien figurer dans notre application. Ce qui sera, certainement, apprécié de la part des utilisateurs puisque la prédiction à long terme de l’évolution de la productivité des développeurs serait possible.



Bibliographie

[B1] : Gareth Dwyer, Jack Stouffer, Shalabh Aggarwal : Flask : Building Python Web Services



Nétographie

[N1] https://fr.wikipedia.org/wiki/Fidelity_National_Information_Services (Dernière visite 05/04/2018)

[N2] <http://ensi-junior-entreprise.com> (Dernière visite 05/04/2018)

[N3] https://fr.wikipedia.org/wiki/Gestion_des_ressources_humaines (Dernière visite 05/04/2018)

[N4] <https://www.ibm.com/watson/fr-fr/talent/> (Dernière visite 26/04/2018)

[N5] <http://sp-expert-retail.com/fr/> (Dernière visite 26/04/2018)

[N6] <https://openclassrooms.com/courses/introduction-au-framework-django/decouvrez-larchitecture-mvt> (Dernière visite 02/05/2018)

[N7] <https://www.kaggle.com/pavansubhasht/ibm-hr-analytics-attrition-dataset> (Dernière visite 16/03/2018)



Liste des acronymes

RH : Ressources Humaines

DRH : Directeur des Ressources Humaines

HTML : HyperText Markup Language

HTTP :HyperText Transfer Protocol

MVT : Model-View-Template

MVC : Model-View-Controller

SGBD : Système de Gestion de Base de Données

SGBDR : Système de Gestion de Base de Données relationnelle

SQL : Structured Query Language

UML : Unified Modeling Language

PHP : Hypertext Preprocessor

BSD : Berkeley Software Distribution

MIT : Massachusetts Institute of Technology

IA : Intelligence Artificielle

FIS : Fidelity National Information Services

IBM : International Business Machines

ORM : object-relational mapping

تلخيص

يندرج هذا العمل ضمن مشروع التصميم و البرمجة للطلبة المهندسين المرسمين بالسنة الثانية في المدرسة الوطنية لعلوم الإعلامية. ويتمثل مشروعنا في استعمال اليات التنقيب في البيانات و الذكاء الاصطناعي لاستخراج مستوى انتاجية المبرمجين اعتمادا على معايير مختلفة. و بإمكان المبرمجين تعرف مستوى انتاجيتهم من خلال تطبيق الالكتروني متوفر على الواب.

:الكلمات الرئيسية التنقيب في البيانات \ الانتاجية \ المبرمج

Résumé

Ce travail s'inscrit dans le cadre du projet de conception et de développement à l'École Nationale des Sciences de l'Informatique.

Notre projet, proposé par l'entreprise Fidelity National Information Services FIS , consiste à créer une application web qui permet aux employés de booster leurs niveaux de productivité au sein d'une entreprise.

Ceci est grâce aux analyses des données des employés qui permettent de prévoir leurs taux performance.

Mots clés : productivité / analyse des données / employé

Abstract

This work makes part of the design and development project at the National School for Computer Studies.

Suggested by the enterprise Fidelity National Information Services FIS, Our project consists in creating a web application that allows employees boosting their productivity level within a company.

This is thanks to employees data analysis that allows to predict their performance rate.

Keywords: Productivity / Data analyses / Employee.