|  |  |  |  |
| --- | --- | --- | --- |
| charge_de_emploi_CMJN |  |  |  |
| **Dossier-Projet** | | |
|  | | |
|  |  | ◢ |
|  |  |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| *Nom de naissance* | ▶ | COURIOL |
| *Prénom* | ▶ | Fred |
| *Adresse* | ▶ | 6, Av. de Villeneuve Saint-Georges  94600 CHOISY LE ROI |
|  |  |

|  |
| --- |
| **Titre professionnel visé** |
|  |
| **Développeur-se logiciel - Niveau III** |
|  |

|  |
| --- |
| **Sommaire** |
|  |

[Remerciements 3](#_Toc485034207)

[Résumé du projet en anglais 4](#_Toc485034208)

[Liste des compétences du référentiel 5](#_Toc485034209)

[Cahier des charges ou expressions des besoins de l’application à développer 6](#_Toc485034210)

[a) Gestion des utilisateurs 9](#_Toc485034211)

[b) Gestion des absences 10](#_Toc485034212)

[c) Circuit de validation des demandes 11](#_Toc485034213)

[d) Echanges par mail 13](#_Toc485034214)

[Spécifications fonctionnelles 17](#_Toc485034215)

[Spécifications techniques 24](#_Toc485034216)

[Réalisations 26](#_Toc485034217)

[a) Etude de la faisabilité 26](#_Toc485034218)

[Conclusions 33](#_Toc485034219)

|  |
| --- |
| Remerciements |
|  |
|  |
| Je tiens à remercier Nicolas HOVART et Emmanuel LE PEVEDIC qui, tout au long de notre cursus, nous ont formés, accompagnés et nous ont apportés un très bon capital de connaissances et de méthodes.  Remerciements à Simplon et à tous ses employés, particulièrement à :   * Laëtitia AMOUROUS pour son implication et son dévouement à nous mettre dans les meilleures conditions. * Sarah GUERRIC qui a contribué à nous faire découvrir React en un temps cours tout en nous captivant. * Elise FRAISSE pour son accompagnement et sa bonne humeur communicative. * Erwann KEZZAR pour ses nombreux conseils et encouragements et ce, depuis le jour des entretiens de sélection. * David OSTERMANN, Eric SIBER, Yvan SRAKA, Medhi MAIZATE, Enzo ESTEVES, Norbert NADIR, Victor HANSEMANN qui ont aussi, chacun à leur manière, amené leurs pierres à notre édifice.   Merci à tous les simploniens car les échanges sur Slack sont une source d’informations et d’inspirations.  Remerciements à La Poste qui a permis que cette aventure ait lieu en particulier à Charles et Nelly qui nous ont chaperonnés.  Mention spéciale à la Promo 1, « les précurseurs », nous avons, par nos différents échanges et entraides, su avancer et grandir ensemble, merci pour cette belle aventure humaine.  Un remerciement particulier à mes collaborateurs de projet, Jean LEFRANCOIS et Mokhtar KHIDER, avec lesquels j’ai eu plaisir à travailler. |

|  |
| --- |
| Résumé du projet en anglais |
|  |
|  |

The goal of our project is to realize an application allowing the management of the absences in a company based on a workflow of e-mail validation.  
  
To insure the functioning, 5 types of different actors intervene each having a distinct role, basically after authentication:

* Any employee can formulate a requirement of absence.
* His team leader can validate or refuse the demand. He can also indicate irregular absences of any employee of his team.
* The Human Resources verify the validity of the demand.
* The administrator manages the database of the employees.
* The department of pay which receives an extraction allowing it to establish pay sheet.

Every employee can consult his calendar as well as that of his team. The graphical representation permits to distinguish every type of absence according to its color.

* The highlighted absences are mainly the paid leaves, RTT, the rest days and the sick leaves.
* The not validated demands will be materialized by a shaded off these colors.

To realize this project, our team of 3 persons developed the pattern Model View Controller. We chose to develop mainly in Java for the Back and React for the Front and in parallel, we established our database in MySQL.

Please note: The Back is built in API Rest and the Front is using Single Page Application principle.

|  |
| --- |
| Liste des compétences du référentiel |
|  |
|  |

* **Rappel des compétences du référentiel :**
* Maquetter une application
* Concevoir une base de données
* Mettre en place une base de données
* Développer une interface utilisateur
* Développer des composants d'accès aux données
* Développer des pages web en lien avec une base de données
* Mettre en œuvre une solution de gestion de contenu ou e-commerce
* Développer une application simple de mobilité numérique
* Utiliser l’anglais dans son activité professionnelle en informatique
* Compétences transversales de l’emploi type :
  + Actualiser et partager ses compétences en développement informatique
  + Organiser son activité en développement informatique

|  |
| --- |
| Cahier des charges ou expressions des besoins de l’application à développer |
|  |
|  |

**L’expression des besoins telle que le client (précisément, notre équipe de formateurs) nous l’a communiquée :**

*La société Simplon veut optimiser sa gestion du personnel, et notamment la gestion des absences. Pour cela, elle veut mettre en place un circuit de validation ainsi qu’un logiciel permettant de valider ce circuit. L’objectif est de mieux gérer son personnel et éviter des sous effectifs.*

*Le salarié se connectera avec son adresse Gmail et un mot de passe. Un premier mot de passe lui sera attribué, qu’il devra modifier à la 1° connexion. Il remplira un formulaire pour sa demande de congés. Un mail sera automatiquement envoyé à son N+1, avec un lien de validation ou de refus des congés. Le salarié sera averti par mail de la réponse. S’il y a refus, la demande sera enregistrée dans la base avec la mention refus et le circuit s’arrête. Si la demande est acceptée, elle sera affichée dans Google Calendar. Le N+1 aura la possibilité de visualiser les congés de ses subalternes sur une période donnée. Le N+1 aura aussi la possibilité de saisir des congés maladies ainsi que des absences irrégulières.*

*Un récapitulatif mensuelle sera générée pour le calcul de la paye.*

*Un mail sera aussi envoyer à la responsable RH pour information et vérification des règles RH.*

*Un SysAdm gérera les différentes bases de données.*

*Sa page d’accueil sera composé d’un récapitulatif de ses congés restant à prendre ainsi que d’un formulaire de demande de congés.*

*Il aura aussi accès à ses informations personnelles qu’il pourra demander à faire modifier par la responsable RH, en cas d’erreurs et à un récapitulatif de ses absences depuis le 1° janvier.*

**Liste des acteurs :**

* **Employé :** La personne qui fait sa demande de congés. Elle peut aussi visualiser le solde de ses différentes absences. Niveau 0.
* **N+1 :** La personne qui valide la demande de congés des personnes sous ses ordres. Elle saisit aussi certaines absences et peut faire une demande de congés. Niveau 1.
* **Secrétaire RH :** La personne qui vérifie d’un point de vue RH, la validité des congés. Elle peut les modifier ou les annulés. Elle peut aussi faire une demande de congés.
* **Administrateur de la base :** Il saisit les nouveaux arrivants et gère les problèmes pouvant survenir.
* **Service paye :** C’est un acteur externe qui reçoit un état mensuel des congés pour établir la paye.

**Liste des absences à prendre en compte :**

* Congé payé
* Congé maladie
* RTT
* Repos compensateur
* Autres cas d'absence
  + Absence irrégulière
  + Congé exceptionnel (raison familiale, fête religieuse, etc…)
  + Congé sabbatique
  + Autres

**Liste des fonctionnalités :**

* Identification
* Administration des employés
* Création de la chaîne de validation
* Demande de congé
* Suivi de la demande
* Validation de la demande à chaque niveau
* Consultation des demandes pour une équipe
* Envoi de mail
* Maj du calendrier google agenda
* Création d’un rapport pour la RH chaque mois.

**Circuit de validation :**

* Le demandeur formule une demande
* L'enregistrement de la demande entraine l'envoie d'un mail de demande contenant un lien sur la décision vers le valideur avec son backup en copie.
* Le valideur ou son backup refuse, un mail de refus est transmis au demandeur et le workflow s'arrête là.
* Le valideur accepte, un autre mail est transmis au RH pour prise en compte.

## Gestion des utilisateurs

Les utilisateurs de l’application doivent s’authentifier à l’aide de leur adresse mail ainsi que de leur mot de passe donnant accès aux environnements auxquels ils ont droit.

Il revient à l’administrateur de l’application de gérer la base des personnes. Il peut ainsi créer, modifier ou supprimer des utilisateurs.

Un utilisateur se définit par son nom, son prénom, son matricule, son adresse mail, son ou ses rôles, l’équipe dont il dépend, le service RH de validation dont il dépend.

En cas de création, le mot de passe, les compteurs de congés sont définis à leurs valeurs par défaut. A noter que tous les utilisateurs ont à minima le rôle de demandeur.

L’administrateur doit aussi gérer les équipes ainsi que les Responsables, essentiel pour permettre l’usage du calendrier d’équipe.

## Gestion des absences

Différents type d’absences sont gérés par l’application, ainsi, un employé peut formuler une demande d’absence selon le cas de type congé payé, RTT, repos compensateur (récupération suite à heures supplémentaires), congé exceptionnel (maternité, paternité, fête religieuse, évènement familial,…).

Un responsable d’équipe ou le service RH peut également déclarer l’absence inopinée d’un employé dont il a la charge. Dans le cas où un employé est en absence irrégulière ou en congé maladie, son responsable ou son service Rh peut signaler l’absence dans l’application pour prise en compte. Il est aussi possible de signaler une absence légitime dans le cadre d’un oubli de l’employé ou d’une régularisation.

Remarque : Les compteurs de congés ou de RTT sont exprimés en jour ce qui n’est pas le cas pour le repos compensateur qui est exprimé en heure.

La pose de congé engendre un décompte en fonction de la durée tronquée des jours travaillés puisque l’application sait faire la différence entre jours travaillés et les jours fériés et les jours de repos hebdomadaire.

A ce propos, les jours reconnus comme fériés, au nombre de 13, sont par défaut jours non travaillés mais selon les conventions propre à l’entreprise, il est possible de définir un ou plusieurs jours fériés comme jours travaillés.

De même, par défaut, les 2 jours du week-end sont considérés comme jours de repos hebdomadaire mais il est possible de redéfinir ces jours de repos hebdomadaire.

## Circuit de validation des demandes

Un employé soumet une demande d'absence par le biais du formulaire prévu à cet effet.

La prise en compte de la demande entraine automatiquement l'envoi d'un mail destiné au valideur et à son backup en copie. Ce mail contient un lien permettant de se prononcer sur l’opportunité de la demande. En parallèle, un mail notifiant la prise en compte de la demande est envoyé au demandeur.

Si la demande est validée, deux mails sont transmis, le premier au demandeur pour lui rendre compte de la validation et le deuxième, destiné au service RH pour prise de décision. De même que pour le responsable hiérarchique, le mail destiné au service RH contient un lien lui permettant de se prononcer.

En cas de validation du service RH, le demandeur, son responsable et son backup en sont notifiés par mail, la base de données est actualisée en conséquence. Une tâche est envoyée à l’agenda du demandeur afin d’y faire figurer la période d’absences.

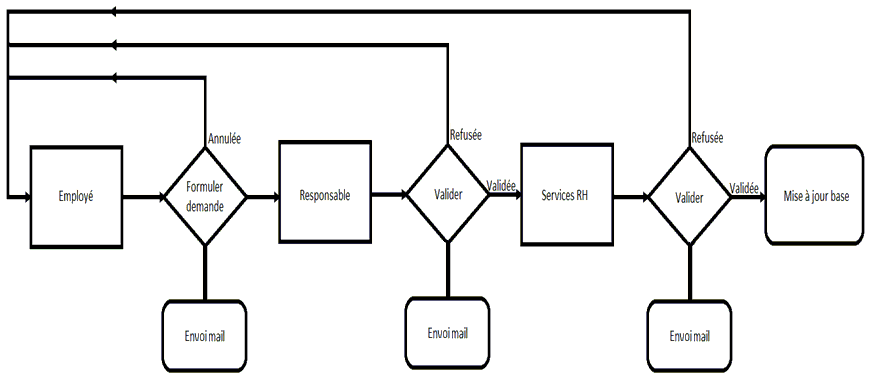
Dans le cas où le valideur ou son backup ou les RH oppose un refus, un mail signifiant le refus et mentionnant le motif du refus est transmis au demandeur. Cela entraîne, par la même occasion, l'arrêt du workflow.

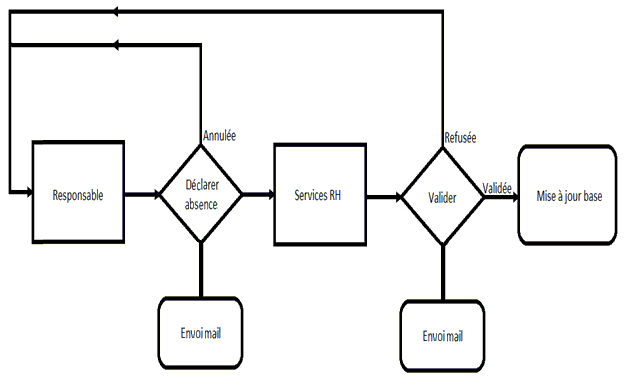
Le demandeur peut à tout moment suivre le workflow de sa demande. Il a aussi la possibilité de relancer sa demande restée en attente de validation.

Il peut aussi annuler la demande en cours de décision.

**Circuit de validation par le schéma :**

**Demande d’absence**



**Déclaration d’absence**

Remarque : Le demandeur est notifié par mail à chaque changement d’état de sa demande. Il peut aussi suivre l’état de sa demande en consultant son tableau des dernières demandes.

## Echanges par mail

**Exemple de mail que doit transmettre l’application :**

De : Gestion des absences <[gestion.absences75@gmail.com](mailto:absence.pt@laposte.fr)>   
Date : 02/06/2017 13:30 (GMT+01:00)   
À : [stephanie.lambert@gmail.com](mailto:stephanie.lambert@gmail.com)

Cc : [lionel.rouviere@gmail.com](mailto:lionel.rouviere@gmail.com)   
  
  
Objet : Demande d’absence N° DEM012768

Bonjour,

L'employé(e) DUPONT Arnaud dont vous êtes le ou la responsable, demande à bénéficier de congés payés du 12/06/2017 au 16/06/2017.

En qualité de responsable, merci de bien vouloir étudier la demande référencée sous le N° [DEM012768](http://xxxxxxxxxx.xx/?avisresp=2E72ddecd0a087a0c730e0e7b090c06) et d'y apporter votre validation ou votre refus **en cliquant sur le lien**.

Nous attirons votre vigilance sur l'importance et votre engagement concernant votre validation ou votre refus.

Cordialement,  
Service de gestion des absences.  
  
  
Ce message vous a été envoyé automatiquement, merci de ne pas répondre.

L’exemple ci-dessus concerne une demande d’absence formulée, soumise à décision du responsable hiérarchique ou de son backup. Le responsable ou son backup, en cliquant sur le lien contenu dans le mail accède au formulaire de décision (sous couvert de l’authentification du décisionnaire).

Nota bene : Le lien du mail propose un accès direct au formulaire de décision, il reste, pour autant, la possibilité d’y accéder par le biais de la liste des dernières demandes.

Tous les mails transmis par l’application doivent être émis à partir de l’adresse mail « [gestion.absences75@gmail.com](mailto:absence.pt@laposte.fr) » qui doit être créé pour le projet.

Ci-dessous, la liste de mails que doit pouvoir transmettre l’application.

**Cas des demandes d’absences :**

* **Demande formulée et soumise par le demandeur (notification de prise en compte)**
  + Destinataire : Le demandeur
* **Demande soumise à décision du responsable hiérarchique ou du backup (avec lien pour décision hiérarchique - voir l’exemple ci-dessus)**
  + Destinataire : Le responsable hiérarchique
  + En copie : Le backup
* **Demande validée par le responsable hiérarchique ou le backup (notification)**
  + Destinataire : Le demandeur
  + En copie : Le responsable hiérarchique
  + En copie : Le backup
* **Demande validée par le responsable hiérarchique ou le backup (avec lien pour décision RH)**
  + Destinataire : Le service RH
* **Demande refusée par le responsable hiérarchique**
  + Destinataire : Le demandeur
  + En copie : Le responsable hiérarchique
  + En copie : Le backup
* **Demande validée par le service RH**
  + Destinataire : Le demandeur
  + En copie : Le Responsable hiérarchique
  + En copie : Le backup
  + En copie : Le service RH
* **Demande refusée par le service RH**
  + Destinataire : Le demandeur
  + En copie : Le Responsable hiérarchique
  + En copie : Le backup
  + En copie : Le service RH
* **Demande annulée par le demandeur (cas d’une demande en cours)**
  + Destinataire : Le décisionnaire en cours
  + En copie : Le demandeur
  + En copie : Le responsable hiérarchique (si décisionnaire est le service RH)
  + En copie : Le backup
* **Demande annulée par le demandeur (cas d’une demande validée à date non échue)**
  + Destinataire : Le demandeur
  + En copie : Le responsable hiérarchique
  + En copie : Le service RH

**Cas des déclarations d’absences inopinées :**

* **Déclaration formulée par le responsable hiérarchique (notification de prise en compte)**
  + Destinataire : Le responsable hiérarchique
* **Déclaration formulée par le responsable hiérarchique (avec lien pour décision RH)**
  + Destinataire : Le service RH
* **Demande validée par le service RH**
  + Destinataire : Le Responsable hiérarchique
  + En copie : L’employé
  + En copie : Le service RH
* **Demande refusée par le service RH**
  + Destinataire : Le Responsable hiérarchique
  + En copie : Le service RH

**Cas particulier de l’envoi d’état mensuel des absences validées :**

* **A date définie, envoi du récapitulatif élaboré par l’application**
  + Destinataire : Le service de paye
  + Pièce jointe : Etat des absences pour la période du mois écoulé

|  |
| --- |
| Spécifications fonctionnelles |
|  |
|  |

L’application de notre projet doit être réalisée sous la forme d’un site internet proposant des pages web interagissant avec une base de données.

L’accès à l’application doit être sécurisé, chaque intervenant doit s’authentifier pour accéder aux pages auxquelles il a droit.

Tous les intervenants de l’application auront, à tout moment, la possibilité de consulter l’aide adaptée en fonction de son rôle.

L’application doit également pouvoir intégrer des fonctionnalités d’envoi de mails dans le cadre du workflow ainsi que d’ajout de planifications dans l’agenda des demandeurs dans le cadre du suivi sur leur mobile.

Le rôle particulier du valideur backup peut être utile en cas de défection du responsable. En l’absence du responsable ou par délégation de celui-ci, le valideur backup peut se substituer au valideur attitré afin de se prononcer sur la décision à donner à la demande qu’il a à sa charge.

Chaque employé a la possibilité de visualiser son calendrier ainsi que celui de son équipe. Les services RH peuvent visualiser le calendrier de l’ensemble des employés de l’entreprise. Dans le cas particulier où l’employé est responsable, en guise de calendrier d’équipe, il visualise à la fois celui de l’équipe qu’il encadre, celui de ses collaborateurs et celui de son propre responsable.

L’application propose une visibilité de 3 mois pour le calendrier individuel et d’un mois et demi pour celui de l’équipe. L’utilisateur peut faire usage des boutons de navigation permettant d’avancer ou de reculer d’un mois.

Chaque type d’absence est distingué par une couleur (qui est estompée pour les demandes en cours et qui le restent tant que la décision finale n’a pas eu lieu).

Les jours de repos hebdomadaire ainsi que les jours fériés sont aussi matérialisés par une couleur distinctive.

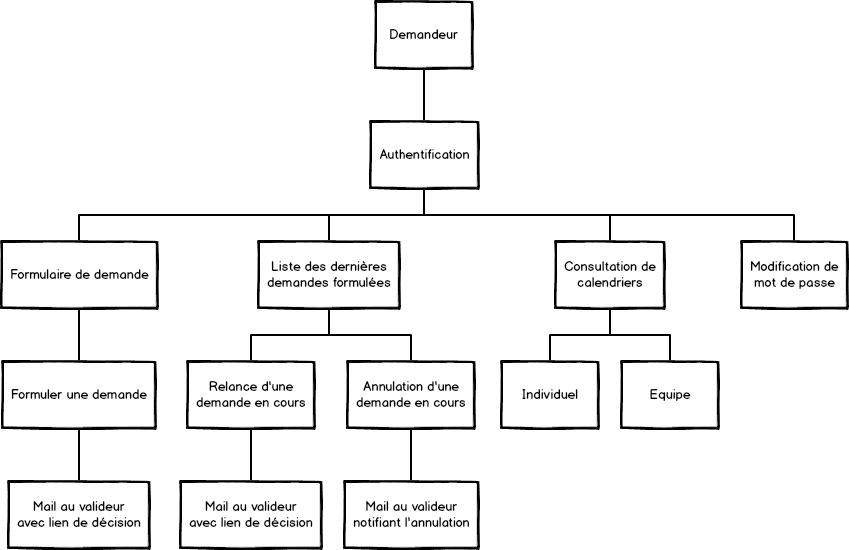
Un employé a possibilité de consulter son calendrier ainsi que celui de son équipe comprenant ses collaborateurs et son responsable.

Le but du calendrier d’équipe est de fournir aux employés une visibilité sur les forces en présence avant de formuler une demande.

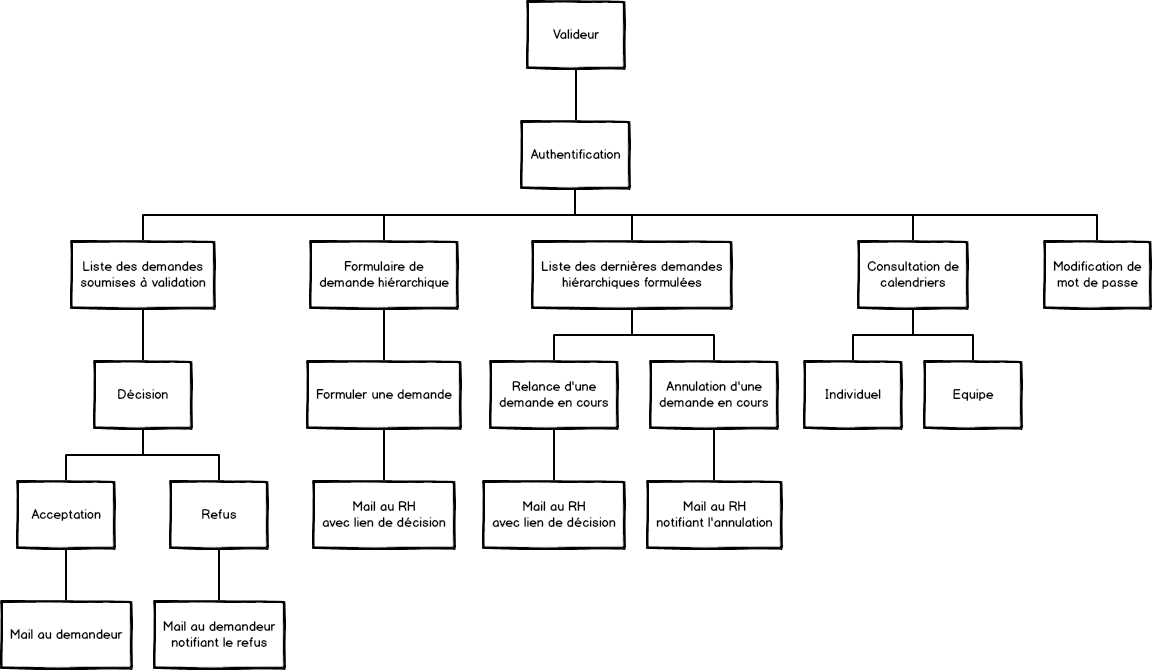
Remarque : L’application doit être en mesure de tenir compte des jours fériés et des jours de repos hebdomadaire permettant ainsi le décompte de jours travaillés lors de la formulation d’absences.

**Représentation du fonctionnement par le schéma**

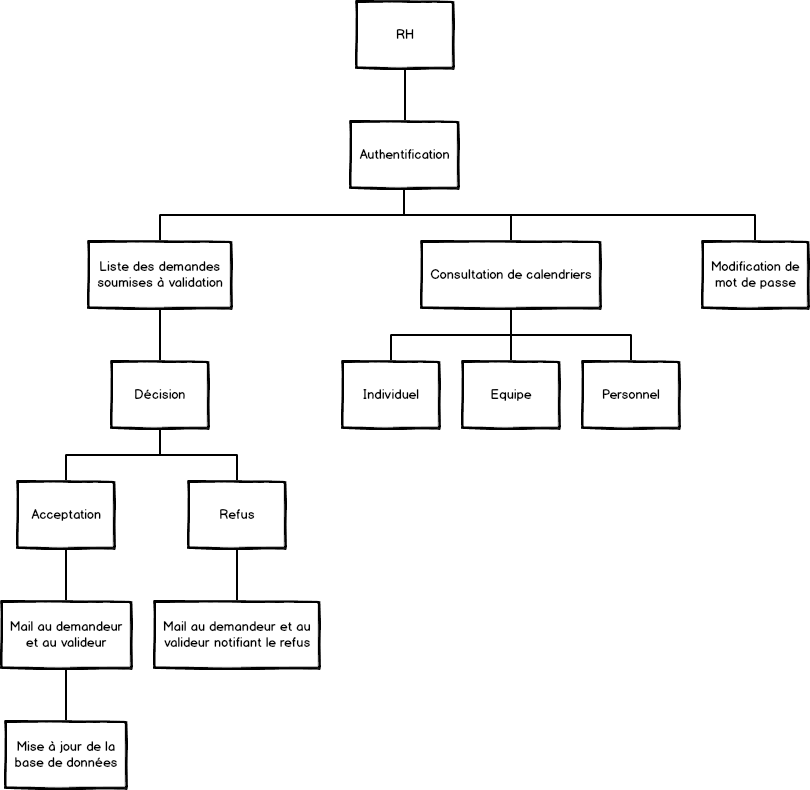
* **Demandeur :**



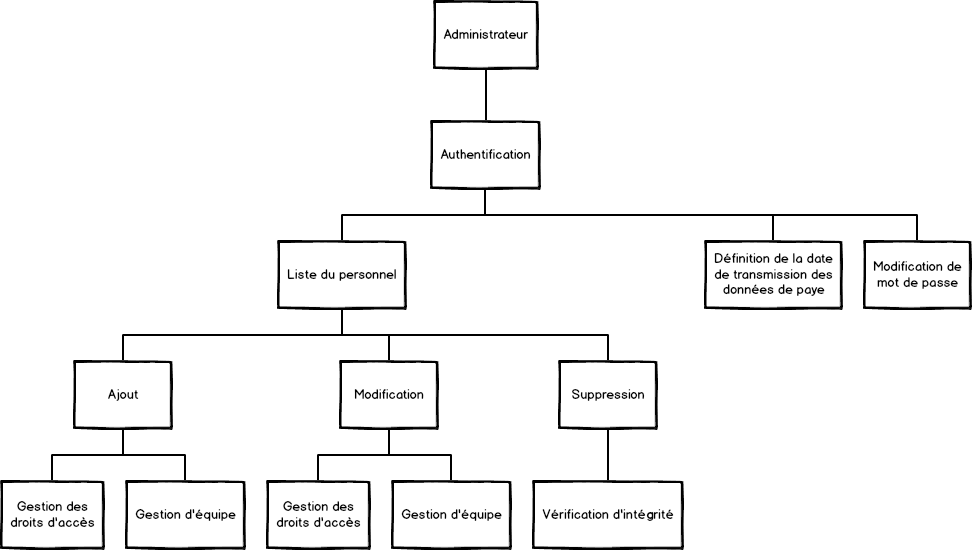
* **Valideur :**



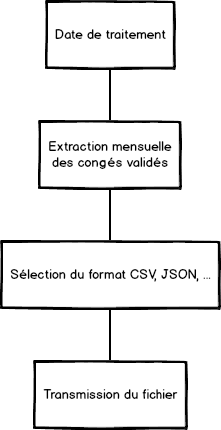
* **Valideur RH :**



* **Administrateur :**



* **Service de paye :**



|  |
| --- |
| Spécifications techniques |
|  |
|  |

Chaque employé doit pouvoir accéder à l’application à partir d’un ordinateur ou d’un mobile, le choix se porte sur un site internet proposant des pages web interagissant avec une base de données.

Les utilisateurs ont accès uniquement aux pages auxquelles ils ont droit parmi les pages suivantes :

* Liste des dernières demandes (tout le monde)
* Nouvelle demande d’absence (tout le monde)
* Déclaration d’absence (le responsable d’équipe)
* Décision du responsable (le responsable d’équipe)
* Décision du service RH (service RH)
* Reliquat de congés dans une équipe (le responsable d’équipe)
* Calendrier personnel (tout le monde)
* Calendrier d’équipe (tout le monde)
* Calendrier de l’entreprise (service RH)
* Gestion du personnel (administrateur)
* Gestion des équipes (administrateur)
* Aide (tout le monde)

Dans cette liste figurent les utilisateurs habilités à accéder à chacune des pages web citées (voir colonne de droite).

Il convient que le champ d’action de certaines pages est limité à l’utilisateur ou à l’équipe de l’utilisateur.

La base de données doit disposer de la liste des tables suivantes :

**Tables fortement évolutives :**

* EMPLOYE recensant les employés de l’entreprise.
* ABSENCE consignant la liste des absences.
* UTILISATEUR en charge des informations d’authentification.

**Tables moyennement évolutives :**

* EQUIPE listant les services ainsi que leur responsable.
* SERVICERH définissant le service RH en charge de l’employé.

**Tables faiblement évolutives :**

* ROLE permettant de définir le domaine d’activité des utilisateurs.
* TYPE relative aux différents types d’absences.
* STATUT relative aux différents statuts des demandes.

A noter que la table EQUIPE doit posséder un lien récursif qui permet, dans le cas d’un responsable, de déterminer l’équipe qu’il encadre, l’équipe dont il est membre et son propre responsable.

Les tables UTILISATEUR et ROLE, permettant de gérer les authentifications et les accès donc à priori accessibles par tous les internautes, doivent contenir le strict minimum nécessaire par mesure de sécurité contre les éventuels piratages de données. Les seules données concernées sont l’adresse mail, le rôle et le mot de passe qui est crypté.

|  |
| --- |
| Réalisations |
|  |
|  |

## Etude de la faisabilité

* **Vision fonctionnelle**

Il s’agit de réaliser l’application de sorte que :

* Le Back et le Front soient en pattern MVC.
* L’application soit associée à une base de données.
* Les vues du Front respecte le principe du Single Page Application.
* **Equipe Projet**

En temps normal, ce type de projet doit faire intervenir plusieurs équipes mais dans la cadre du projet et de notre formation, notre équipe n’est constituée que de 3 apprenants qui endossent les casquettes de tous les acteurs censés intervenir de la demande du client jusqu’à la livraison de l’application. Cela a pour avantage de raccourcir le temps de réalisation qui est de ce fait tronqué des éventuelles échanges qu’aurait engendrées des réunions de calage en amont ainsi que les arbitrages qui pourraient découler des potentiels désaccords.

En résumé, pour notre équipe composée de 3 apprenants, il nous suffit d’être d’accords entre nous quitte à procéder à un vote où le minima de 2 contre 1 permet d’entériner certaines décisions qui ne recueillent pas l’adhésion générale.

* **Délai de réalisation**

Le temps imparti pour réaliser l’application est estimé à 2 mois entrecoupés par nos différentes occupations dans la partie professionnelle dans le cadre de notre alternance.

* **Budget alloué**

Le contexte de notre formation et le fait de la situation multi-casquettes font que le budget alloué pour cette réalisation est facilement estimé à 0€.

* **Vision technique**

La base de données est construite en MySQL workbench à partir de la constitution du diagramme de classes.

Le Back est développé en Java JEE7 associé au framework Spring-Boot embarquant les dépendances essentielles ‘’Hibernate’’, ‘’mysql-connector-java’’ et ‘’spring-boot-starter-web’’. Ce choix a été motivé dans la perspective de faciliter les échanges entre le Back et la base de données d’une part et entre le back et le Front de l’autre.

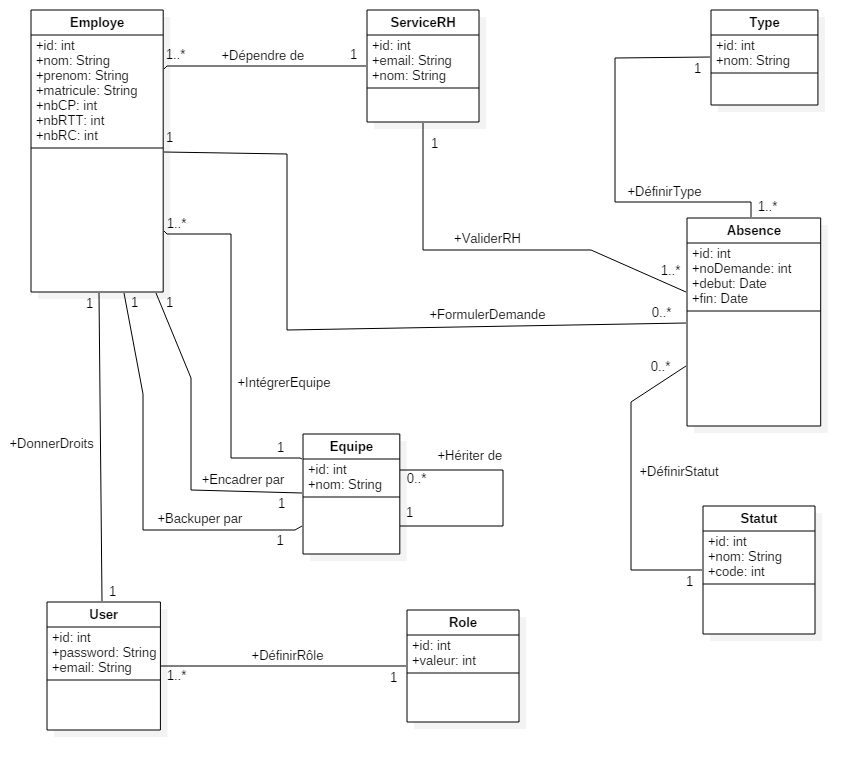
Le Front est développé en React associé à HTML et JavaScript.

Pour les besoins de l’application, il faut implémenter les API suivantes :

* **Côté Back**
  + JavaMail afin de permettre les émissions de mails.
  + Google Calendar pour la création de tâches dans l’agenda des employés.
  + Spring-Security pour la gestion des authentifications et le cryptage/décryptage des mots de passe.
* **Côté Front**
  + jQuery pour l’affichage des calendriers.
  + Bootstrap pour la mise en forme des éléments de chacune des pages.
* **Outils à utiliser**
  + **Suivi du projet**
    - Gestion et répartition des tâches : Trello.
    - Outils de communication : Lync, Lync Attendee et Slack.
  + **Stockage collaboratif**
    - Stockage et partage des sources : Github.
  + **Conception et maquettage**
    - Diagrammes de classes : StarUml.
    - Diagrammes de séquences : StarUml, Balsamiq.
    - Diagrammes d’utilisation : StarUml.
    - Mockups : Balsamiq.
  + **Réalisation**
    - Base de données : MySql Workbench.
    - Connexion à la base de données : JDBC et MySql connector.
    - EDI pour le Back : Eclipse.
    - EDI pour le Front : Atom, Sublim text.
  + **Développement**
    - Langages de programmation : Java, React, HTML, CSS, JavaScript.
    - Framework : Spring-Boot, Spring-Security, Maven, Junit, Bootstrap, jQuery, Hibernate.
    - Création de la base de données : MySQL.
* **Conception générale**

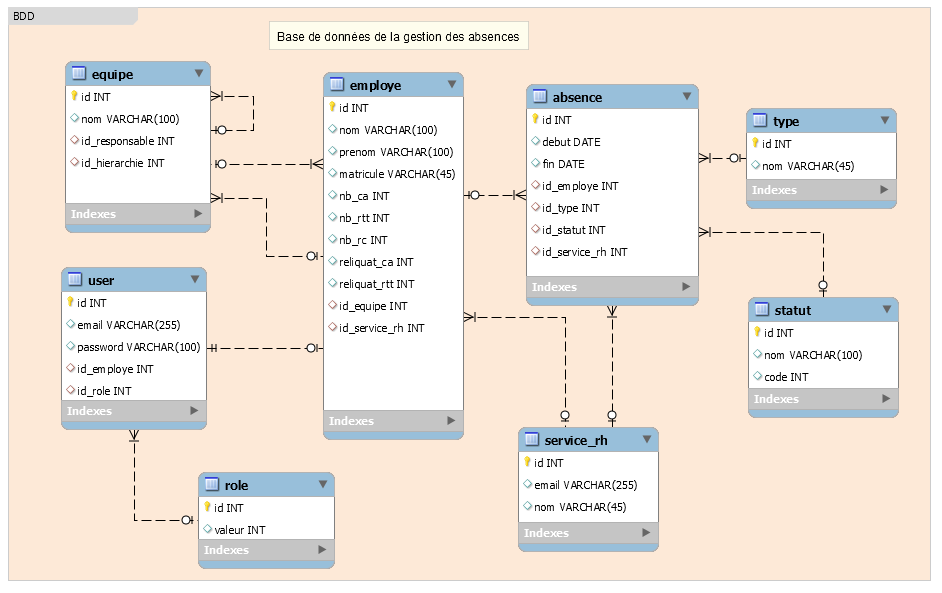
**Diagramme de classes**

En tenant compte du cahier des charges, le diagramme des classes adapté aux besoins de l’application, tel que nous la concevons, s’établit comme suit :



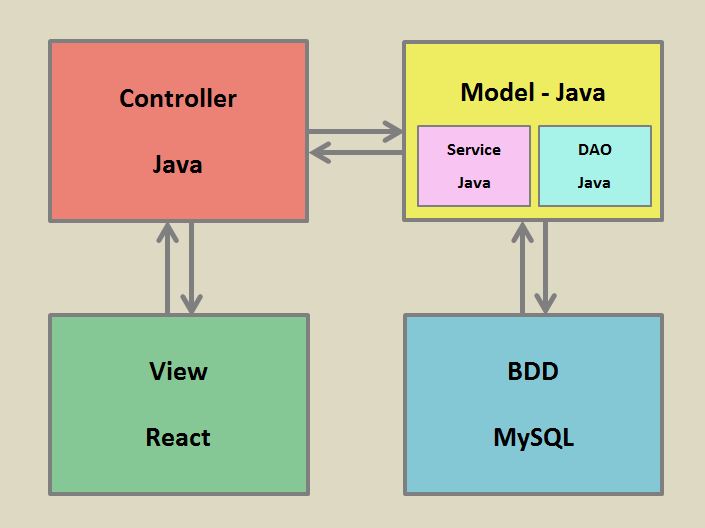
**Diagramme de données**

Le diagramme de classes entraine le diagramme de données suivant :



* **Infrastructure détaillée**

L’architecture de l‘application



|  |
| --- |
| Conclusions |
|  |
|  |