

This document is the intellectual property of its author. Any reproduction, partial or total, without explicit written permission will be considered a copyright infringement.

employee-leave-management-system

Made by: Meradji Anais and Ait Ali Belkacem Sonia

Ceci est une partie modifiée d'un
mémoire de 70 pages. Les données de
congés affichées sont fictives.

3.1 Introduction

Afin de réaliser notre application, nous avons eu recours à certains outils et nous avons opté pour un langage de programmation ainsi qu'un framework adapté.

Notre application est structurée selon une architecture précise et ce afin de garantir l'acquisition d'une scalabilité informatique ainsi qu'une facilité de maintenance. Dans ce dernier chapitre, nous procéderons à la présentation de l'architecture de notre application après avoir évoqué le langage ainsi que le framework utilisé.

Au vu de l'importance d'assurer l'aspect sécurité du système, nous énumérerons les fonctionnalités mises en place afin de servir ce but. Puis nous présenterons les interfaces de notre application.

3.2 Environnement de développement

L'**annexe 4** contient des informations relatives aux machines utilisées lors de la réalisation de ce projet.

3.2.1 Le langage de programmation utilisé

Pour le langage de programmation nous avons utilisé le langage **Java EE**, également connu sous le nom de Jakarta EE. C'est une plateforme de développement Java destinée à la création d'applications d'entreprise. Elle fournit un ensemble de spécifications et de technologies pour le développement, le déploiement et la gestion d'applications d'entreprise robustes, évolutives et sécurisées.

Le frontend, réalisé à l'aide du langage XHTML, est relié à des controllers Java.

Les caractéristiques du langage ainsi que les points forts du langage sont mentionnés au niveau de l'**annexe 4**.

Les outils utilisés > **annexe 5**.

3.2.2 Frameworks

Un framework est une structure logicielle qui fournit un ensemble d'outils, de conventions, de bibliothèques et de composants prédéfinis pour le développement d'applications.

L'utilisation d'un framework peut s'avérer très intéressante puisqu'elle permet un gain de temps lors de la programmation mais aussi un renforcement de la sécurité du système puisque de nombreux frameworks intègrent des fonctionnalités de sécurité prédéfinies, contri-

buant ainsi à réduire les vulnérabilités courantes.

Nous avons opté pour une bibliothèque.

PrimeFaces : une bibliothèque de composants d'interface utilisateur pour le développement d'applications web Java EE. Elle fournit une vaste gamme de composants prêts à l'emploi, tels que des boutons, des champs de saisie, des tableaux etc.

3.3 Architecture MVC

MVC (Model-View-Controller) est un modèle de conception logicielle couramment utilisé pour mettre en œuvre des interfaces utilisateur, des données et de la logique de contrôle. Il met l'accent sur une séparation entre la logique d'affaires du logiciel et l'affichage.

Separation of Concerns (séparer les responsabilités de différentes parties de l'application) ce qui prévoit une meilleure division du travail et une amélioration de l'entretien et rend le code plus modulaire, réutilisable et facile à maintenir.

Voici les trois modules de l'architecture MVC :

- **Modèle (Model)** : gère la logique métier ainsi que la gestion des données qui sont stockées, la manière dont ces données sont manipulées, traitées et présentées dans l'application. De plus, il fournit une interface d'action au contrôleur.

- **Vue (View)** : est chargée de rendre les données du modèle dans une présentation visuelle et interactive pour l'utilisateur final, et affiche les données fournies par le contrôleur. La vue est généralement passive.

- **Contrôleur (Controller)** : synchronise les interactions entre la vue et le modèle dans une application. Il reçoit les demandes de l'utilisateur via la vue, exécute les actions métier appropriées dans le modèle, puis prépare les données pour les transmettre à la vue.

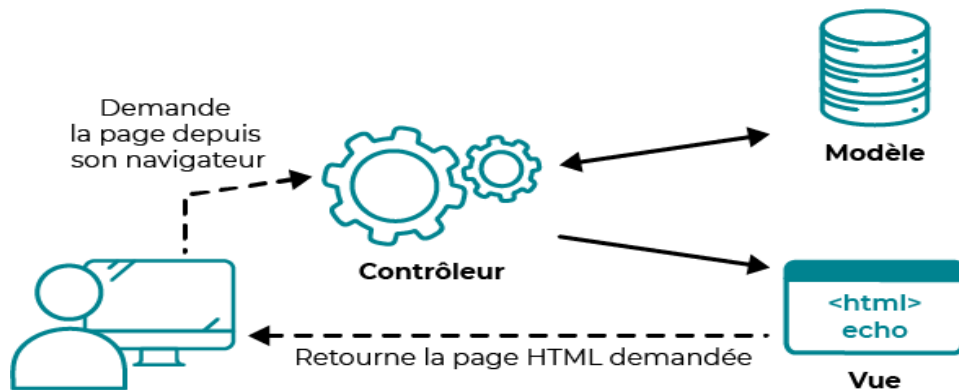


FIGURE 3.1 – L'utilisateur et l'architecture MVC

3.4 Aspects de sécurité

Le côté sécurité de notre application est d'une importance capitale puisqu'il permet de garantir la protection des données sensibles des utilisateurs et d'assurer la conformité réglementaire, en voici quelques-uns des principaux aspects :

-Gestion de l'authentification et de l'autorisation : java EE propose un système d'authentification et d'autorisation robuste via le mécanisme de gestion des identités et des rôles (Identity and Access Management). Il nous permet de définir des contraintes de sécurité dans les descripteurs de déploiement (deployment descriptors) pour restreindre l'accès aux ressources en fonction des rôles et

de l'identité des utilisateurs.

-JCA (Java Cryptography Architecture) : java EE intègre une architecture de cryptographie complète qui permet aux développeurs d'utiliser des algorithmes de chiffrement standard pour sécuriser les données sensibles, notamment en utilisant des API telles que Java Cryptography Extension (JCE) pour le chiffrement et la génération de signatures ainsi que la gestion des clés.

-Sécurité des mots de passe : cette sécurité est assurée à travers l'utilisation de fonctions de hachage qui génèrent un mot de passe haché agissant comme une empreinte numérique unique pour les données, ce qui permet de vérifier l'intégrité des données.

Toute modification ou altération peut être détectée en comparant la valeur de hachage des données reçues avec la valeur de hachage générée au départ.

Il est à noter qu'il s'agit d'une fonction à sens unique, c'est à dire qu'il est impossible de faire de l'ingénierie inverse et de retrouver le mot de passe original à partir de sa forme hachée.

3.5 Présentation des interfaces

Ci-joint le rendu final des différentes interfaces que comporte notre système de gestion des congés pour **Air Algérie**.



FIGURE 3.2 – interface d’authentification



FIGURE 3.3 – interface d’authentification en cas d’erreur

Préférences

i
Choisissez un mot de passe avec minimum 8 caractères, contenant au moins un(e) majuscule, minuscule, chiffre et caractère spécial.

Ancien mot de passe *

Ancien mot de passe...

Nouveau mot de passe *

Nouveau mot de passe...

Confirmation du mot de passe *

Confirmation du mot de passe...

Annuler

Enregistrer

FIGURE 3.4 – interface de changement de mot de passe

Nous allons montrer les interfaces d'un responsable RH qui a donc la visibilité sur la direction dont il est responsable. Pour cet exemple, c'est un responsable de la DSI.

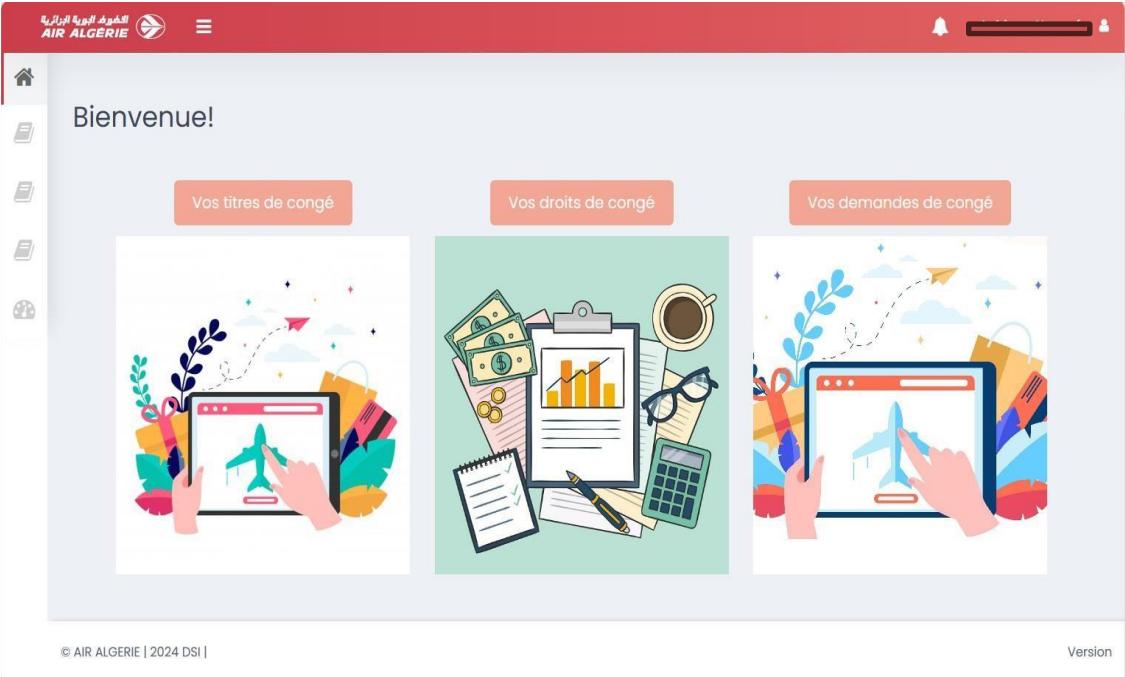


FIGURE 3.5 – interface d'accueil du responsable ressources humaines

Matricule	Exercice	Date début	Date fin	Durée
37999	2023-2024	2024-04-01	2024-04-11	10
37999	2022-2023	2023-08-12	2023-08-15	3
37999	2021-2022	2022-05-01	2022-05-12	11
37999	2020-2021	2021-12-02	2021-12-12	10
37999	2019-2020	2019-07-15	2019-07-20	5

© AIR ALGERIE | 2024 DSI |

Version

FIGURE 3.6 – interface titres de congé



FIGURE 3.7 – interface droits de congé

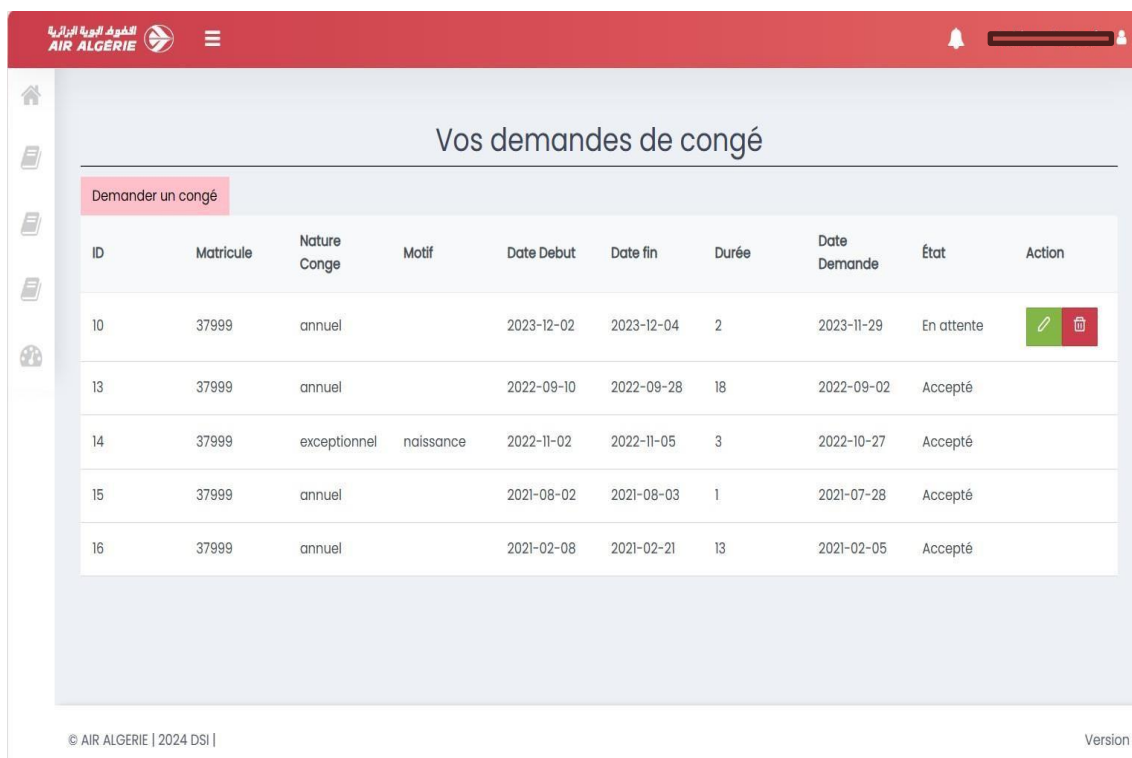


FIGURE 3.8 – interface vos demandes de congé

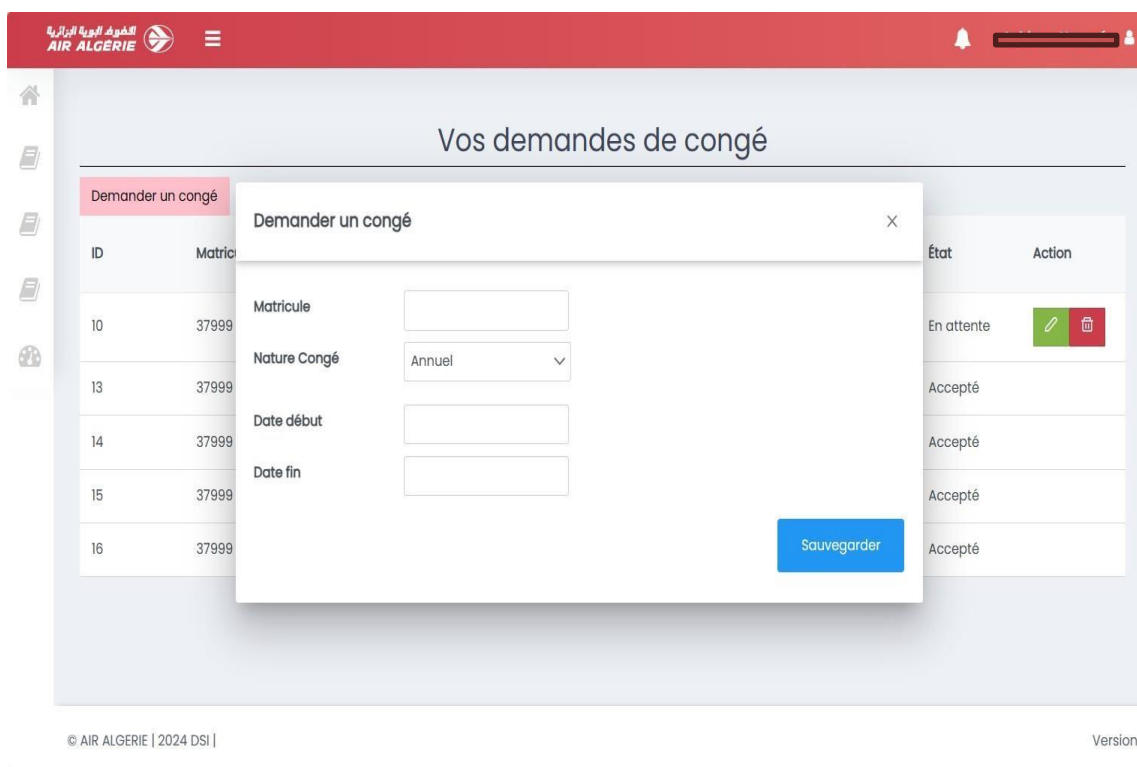


FIGURE 3.9 – interface demander un congé

Ajout d'un droit de congé

X

Matricule

Exercice

Sélectionner

▼

Nb Jours Consommés

Nb Jours Restants

Sauvegarder

FIGURE 3.11 – interface ajout d'un droit de congé

3.6 Conclusion

Ce dernier chapitre nous a permis de donner un aperçu de l'envers du décor puisque nous avons présenté les différents outils, langages et frameworks utilisés avant d'apporter certains détails et clarifications concernant l'architecture adoptée et les aspects de sécurité.

Ce chapitre s'est clos par la présentation des interfaces, marquant ainsi la fin de ce rapport dont les chapitres ont englobé tout le cycle du développement de l'application.

CONCLUSION GÉNÉRALE

Tout au long de la préparation de notre projet de fin d'études, nous avons essayé de mettre en pratique les connaissances acquises durant notre cursus universitaire et avons eu l'occasion de les consolider et d'acquérir, par la même occasion, une certaine aptitude à résoudre des problèmes que les entreprises peuvent rencontrer .

Ce projet nous a permis d'avoir une approche complète du développement d'un système car le plan de travail adopté a englobé les différentes étapes nécessaires lors de réalisation d'une application, à savoir l'étude de l'existant, l'analyse des besoins, la conception ainsi que l'étape réalisation qui en elle même nous a apporté son lot de bénéfices et ce à travers l'acquisition de compétences techniques et l'apprentissage de nouvelles technologies telles que l'utilisation du langage **Java** dans le contexte du dev web ainsi que l'utilisation de My Sql Server pour la réalisation de la base de données.

Faute de temps, nous avons dû réaliser certaines fonctionnalités à la hâte, néanmoins, le résultat nous a globalement plu et les améliorations à apporter demeurent désormais plutôt faciles à implémenter

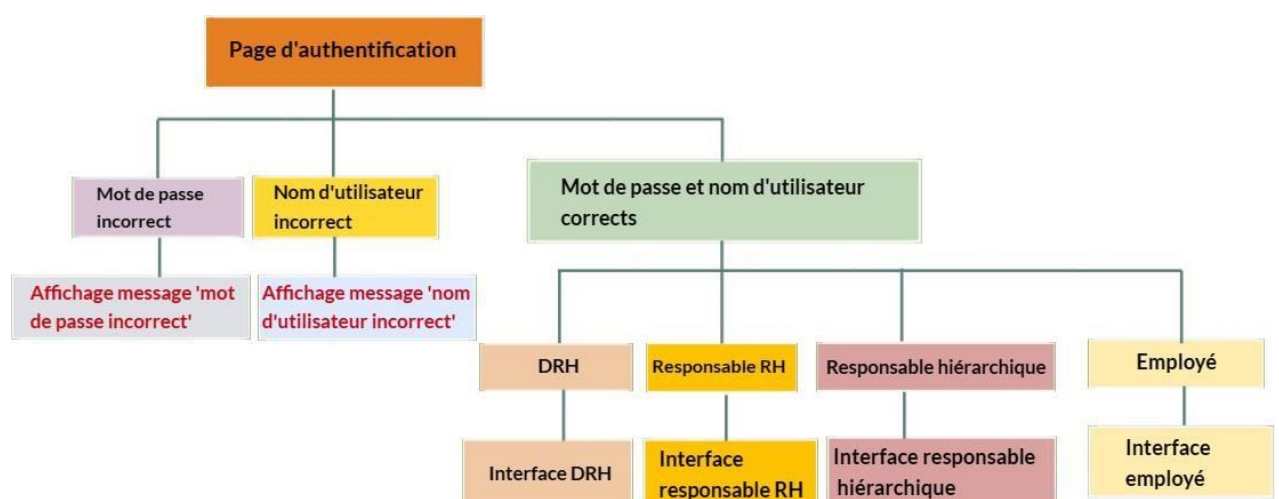
puisque le plus gros du travail est fait.

Ce fut donc une expérience enrichissante qui nous a apporté une meilleure compréhension des défis pratiques ainsi qu'une expérience concrète dans la résolution de problèmes informatiques, ce qui a attisé notre curiosité intellectuelle et a consolidé notre désir d'exercer notre futur métier dans le domaine de l'informatique.

Arborescence de l'application web

Nous présentons ci-dessous l'arborescence de notre application web qui consiste en un schéma organisant les pages de l'application en rubriques.

-Le schéma ci dessous représente la page d'authentification.



Arborescence de l'application web avant authentification

Le diagramme d'activités permet de représenter graphiquement le flux de contrôle entre différentes activités dans un système, nous avons donc décidé de réaliser deux diagrammes d'activités afin d'apporter une représentation visuelle du déroulement des cas d'utilisation de demande ainsi que de traitement des congés.

Diagrammes d'activités

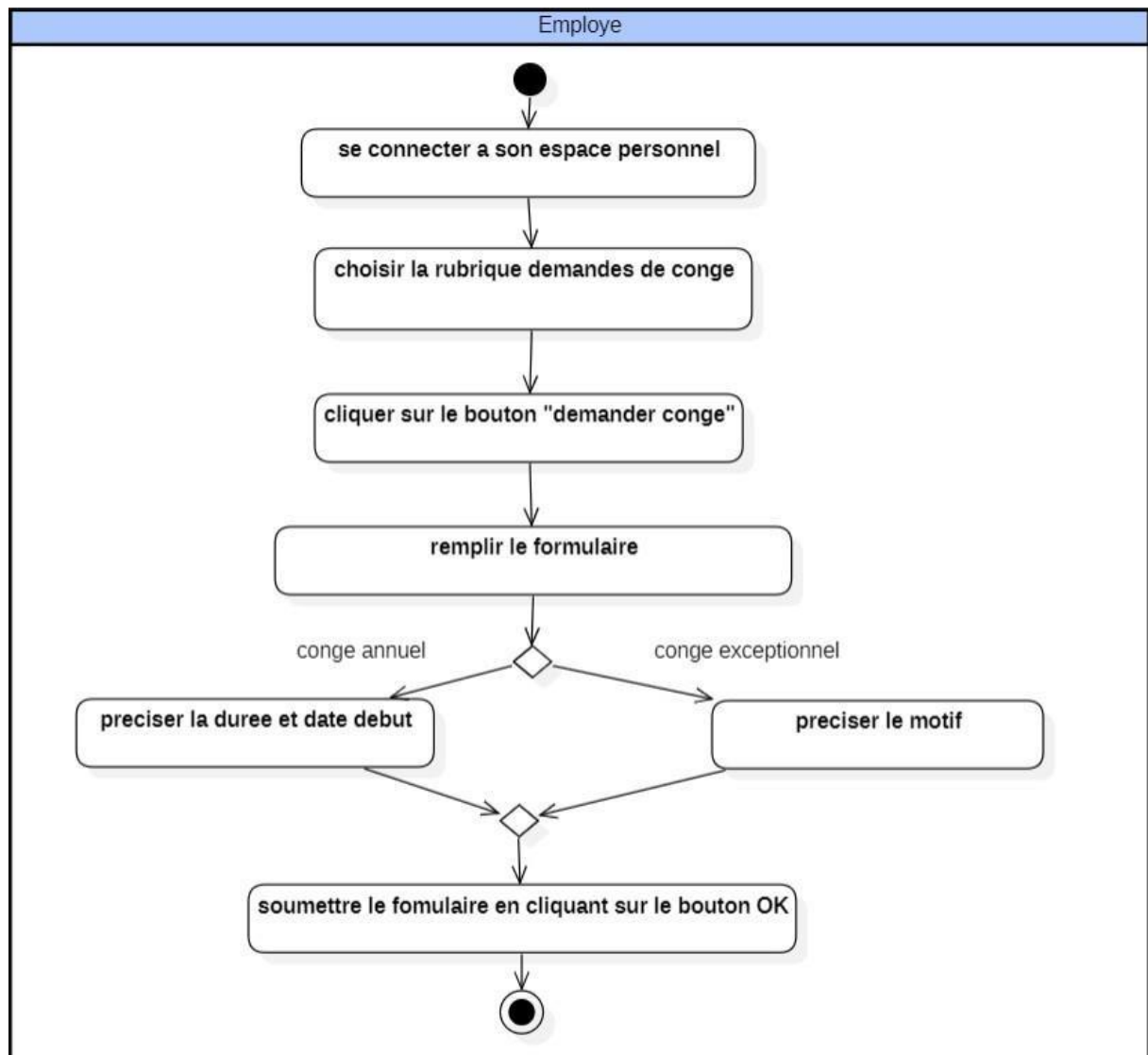


Diagramme d'activité demander un congé

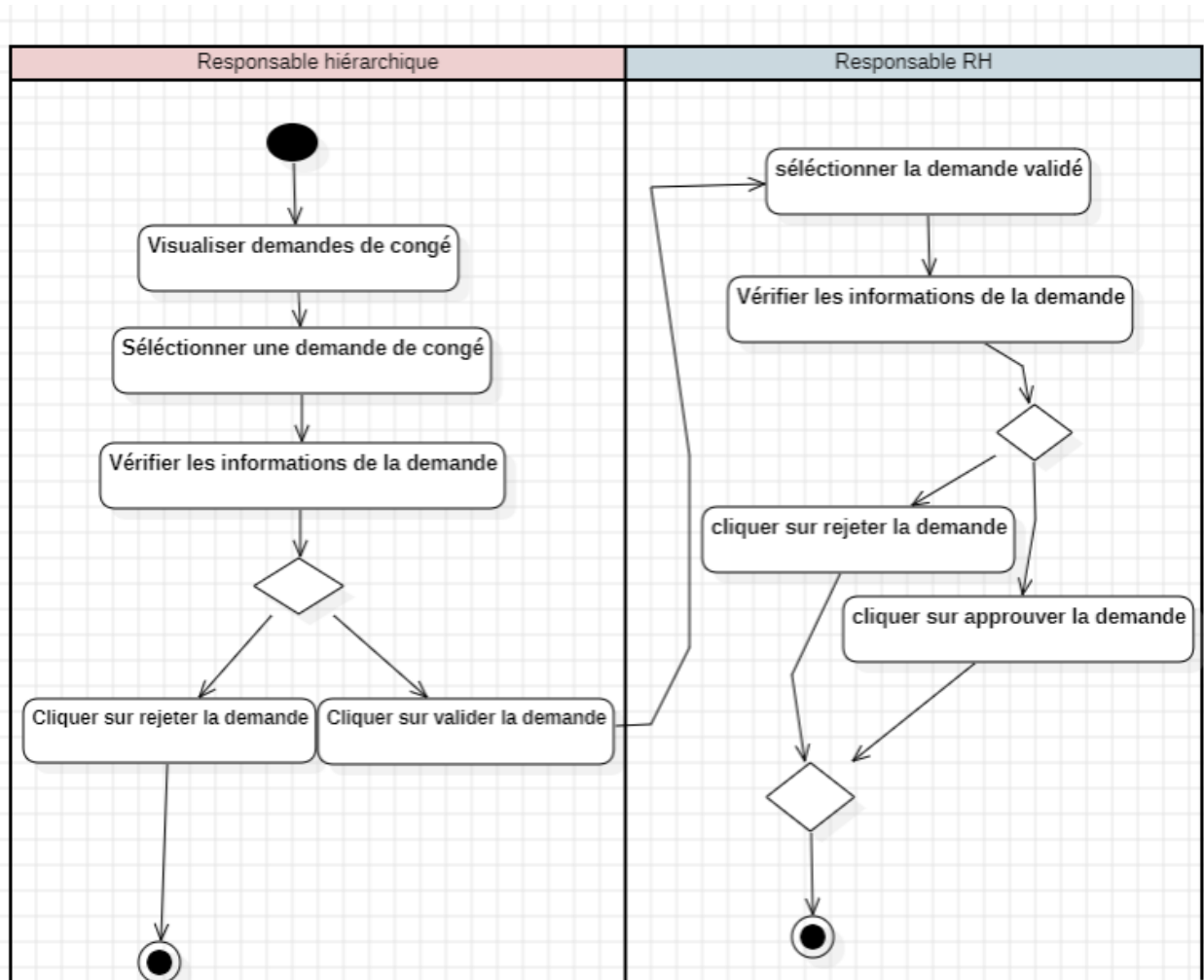


Diagramme d'activité traiter une demande de congé

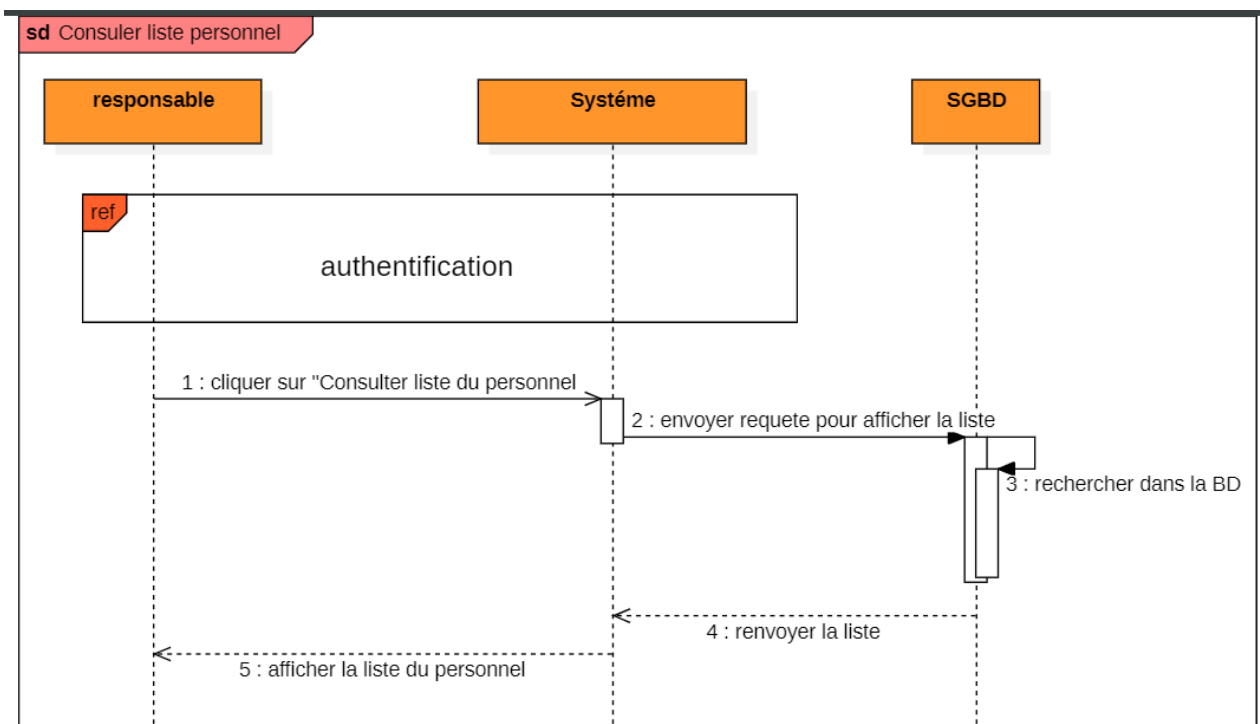


Diagramme de séquence visualiser liste du personnel

Le matériel utilisé

Nous avons utilisé deux machines pour développer notre système :

Ordinateur 1 :

Marque	Microsoft
Processeur	Intel(R) Core(TM) i5-1035G1 CPU @ 1.00GHz 1.19 GHz
Mémoire	8,00 Go
Système d'exploitation	Windows 11

Spécifications de l'ordinateur 1

Ordinateur 2 :

Marque	Lenovo
Processeur	Intel(R) Core(TM) i5-6300U CPU @ 2.40GHz 2.50 GHz
Mémoire	8,00 Go
Système d'exploitation	Windows 10 pro

Spécifications de l'ordinateur 2

Informations concernant le langage utilisé

Caractéristiques clés du langage Java EE : java EE est un langage particulier.

Ci dessous les points clés qui le caractérisent :

-Servlets et JSP (JavaServer Pages) : Java EE inclut des technologies telles que les Servlets et les JSP pour le développement d'applications web dynamiques. Les servlets sont des composants Java qui s'exécutent côté serveur et gèrent les requêtes HTTP, tandis que les JSP permettent de créer des pages web dynamiques en combinant du code Java avec du contenu HTML.

-WebSockets : Java EE propose une API pour la communication bidirectionnelle en temps réel entre le navigateur et le serveur, ce qui permet de créer des applications web interactives telles que les chats en ligne et les tableaux de bord en temps réel.

-Intégration avec les Bases de Données : Java EE offre des API pour l'intégration avec les bases de données relationnelles, telles que JDBC (Java Database Connectivity) et JPA (Java Persistence API). Ceci facilite le développement d'applications web qui interagissent avec des bases de données.

-Déploiement sur serveur d'applications : Les applications Java EE peuvent être déployées sur des serveurs d'applications compatibles, tels que Apache Tomcat, Payara server (celui que l'on a utilisé) et IBM WebSphere, ce qui permet de bénéficier de fonctionnalités telles que la gestion des ressources, la montée en charge et la haute disponibilité.

Points forts du langage Java EE : afin de justifier l'utilisation de Java EE pour le développement de notre application, nous allons présenter les points forts de ce dernier.

-Portabilité : Les applications Java EE sont hautement portables, ce qui signifie qu'elles peuvent être déployées sur une grande variété de plates-formes matérielles et logicielles, grâce à la machine virtuelle Java (JVM).

-Sécurité : La plateforme offre des fonctionnalités de sécurité avancées, telles que l'authentification, l'autorisation, le chiffrement et la gestion des certificats, pour protéger les applications contre les attaques et les accès non autorisés.

-Productivité du développeur : Java EE propose un large éventail de composants et de frameworks prêts à l'emploi, ainsi que des outils de développement avancés, ce qui permet aux développeurs de créer des applications rapidement et efficacement.

-Communauté active : Java EE bénéficie d'une vaste communauté de développeurs, ce qui signifie qu'il existe de nombreuses ressources, forums et bibliothèques disponibles pour aider les développeurs à résoudre les problèmes et à améliorer leurs compétences.

Les outils utilisés

Nous avons utilisé les outils suivants :

Lucidchart : une application de création de diagrammes et de schémas en ligne. Nous avons opté pour LucidChart car la possibilité de collaboration sur un seul et même diagramme nous a permis d'avancer plus rapidement et par conséquent d'être plus efficaces durant la phase conception de l'application.

Il est à noter que nous avons également eu recours à **Star UML** pour la réalisation de certains diagrammes.



Logo de Lucid-chart

Git/Github : Git est un système de contrôle de version, c'est-à-dire un outil qui permet de suivre les modifications apportées aux fichiers d'un projet informatique au fil du temps.

Github permet l'hébergement des projets Git.

Cela nous a grandement aidés à collaborer et à travailler sur différentes fonctionnalités.



Logo de Github

Netbeans : un environnement de développement intégré (IDE) open-source, modulaire et polyvalent.

NetBeans est largement utilisé pour le développement d'applications Java, y compris les applications de bureau, les applications web (Java EE), les applications mobiles (Android), et plus encore.



Logo de Netbeans

My SQL server : MySQL Server est un système de gestion de base de données relationnelle open-source. Il offre une plateforme fiable pour stocker, organiser et récupérer des données, souvent utilisé dans le développement d'applications web.



Logo de MySql server

Payara server : un serveur d'application Java open-source.

Il est à noter que Payara Server est compatible avec la plateforme Java EE, ce qui signifie qu'il prend en charge le développement et le déploiement d'applications d'entreprise Java conformes aux normes Java EE.



Logo de Payara
server