



Groupe 2

RAPPORT DE PROJET

Application Client-serveur pour
traitement OCR

SOMMAIRE

INTRODUCTION

- I. Présentation du projet**
- II. Objectifs du projet**
- III. Acteurs du projet et répartition des tâches**
- IV. Fonctionnalités de l'application**
- V. Durée d'exécution du projet**

CHAPITRE 1 : ETUDE CONCEPTUELLE

- I. Diagramme de cas d'utilisation**
- II. Diagramme de séquences**
- III. Diagramme de classes**

CHAPITRE 2 : IMPLEMENTATION TECHNIQUE

- I. Environnements et technologies de développement**
- II. Présentation de l'application**
- III. Défis et limites de l'implémentation**

CONCLUSION GENERALE

LA BIBLIOGRAPHIE, WEBOGRAPHIE

INTRODUCTION

I. Présentation du projet

Dans un monde de plus en plus digitalisé, l'extraction automatique de données à partir d'images est devenue essentielle pour améliorer l'efficacité et la productivité des entreprises. Notre application web se distingue par sa capacité à charger une ou plusieurs images et à extraire automatiquement du texte, des données de codes-barres, et des données de QR codes. Conçue avec les technologies modernes React.js pour le front-end et Spring Boot pour le back-end, notre solution offre une expérience utilisateur optimale et une robustesse technique inégalée.

II. Objectifs du projet

- Développer une interface utilisateur conviviale pour sélectionner des images depuis la galerie.
- Intégrer un moteur OCR robuste pour reconnaître le texte contenu dans les images.
- Exportation de code-barres et code QR qui conduit vers un lien Web, afficher un texte, ouvrir une application, etc.

III. Acteurs du projet et répartition des tâches

Acteurs du projet

Voici une liste exhaustive des membres du projet :

- **DIABY ABOUBACAR SIDICK** @AboubakarRep
- **MOISE KOUASSI KONAN** @moisekonan

- **MOUNET ANDRE ARMEL RADOINE** @Andrearmel
- **SAWADAGO ZAKARIA** @SawadogoZakaria

Répartition des taches

Nom et prénom	Répartition des taches
DIABY Aboubacar Sidick	Charger une image pour extrait les données texte / code-barres ou code QR
KOUASSI Konan Moise	Récupérer le mot de passe oublier pour l'utilisateur
MOUNET André Armel Radoine	Consulter l'historique de traitement d'images pour l'utilisateur connecté
SAWADAGO Zakaria	Authentification

IV. Fonctionnalités de l'application

- Charger une image pour extrait les données texte :
 - Sélectionner une image depuis votre galerie pour la reconnaissance de texte à partir des images.
- Lecture du code-barres ou code QR :
 - Sélectionner un code-barres ou code QR depuis votre galerie pour afin d'être conduit vers un lien Web, afficher un texte, ouvrir une application, etc.

- Consulter l'historique de traitement d'images pour l'utilisateur connecté :
 - seul l'utilisateur inscrit peut voir ses images convertir en texte et code-barres et code QR qui l'ont dirigé vers un lien Web, afficher un texte, ouvrir une application, etc.
 - L'utilisateur a la possibilité de supprimer les images.
- Authentification :
 - L'utilisateur peut se connecter, s'il est une fois inscrit.
 - L'utilisateur a la possibilité de se connecter avec son compte Gmail.
- Récupérer le mot de passe oublier pour l'utilisateur :
 - A travers son compte Gmail, l'utilisateur peut récupérer son mot de passe.
- Consulter historique des utilisateurs par l'administrateur :
 - L'administrateur peut consulter l'historique de tous les utilisateurs.
 - L'administrateur a la possibilité de supprimer ou télécharger les images convertir en texte, les codes-barres ou codes QR traité par les utilisateurs.

V. Durée d'exécution du projet

Nous reconnaissons l'importance de respecter des délais précis pour le développement et la livraison du projet " OCR".

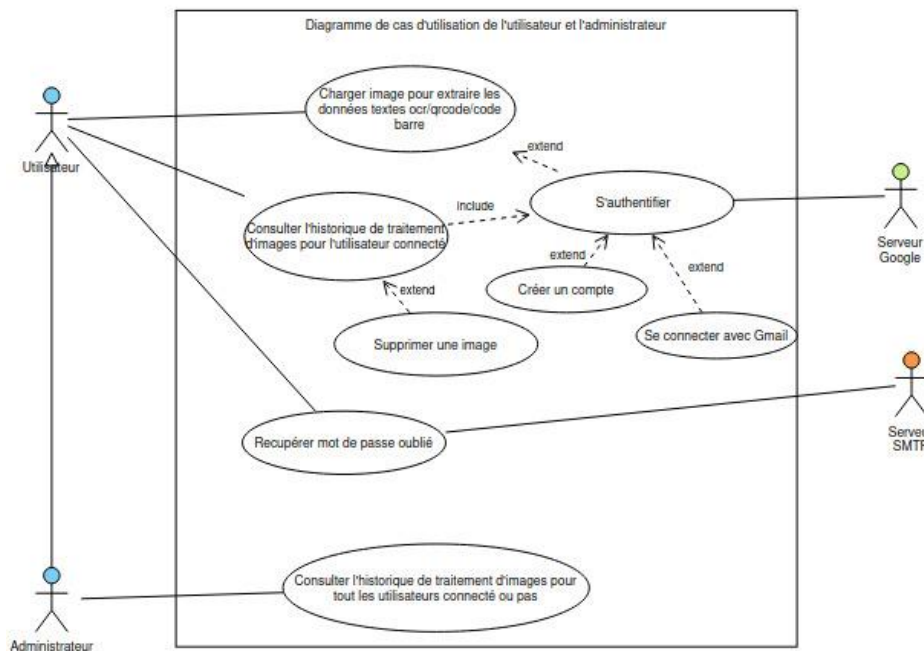
Tache	Avril			Mai				Juin				
	Semaines			Semaines				Semaines				
	2	3	4	5	1	2	3	4	1	2	3	4
Analyse	X											
Conception		X	X	X								
Développement					X	X	X	X	X			
Tests										X	X	
Déploiement												X

Etapes	Durée (semaines)
Analyse	1
Conception	3
Développement	5
Tests	2
Déploiement	1
Total	11

CHAPITRE 1 : ETUDE CONCEPTUELLE

Pour modéliser notre application, nous avons choisi UML pour sa standardisation, sa richesse de notation, et sa flexibilité. Contrairement à Merise, UML est mieux adapté aux méthodologies agiles et aux architectures modernes comme les micro-services. Nous allons maintenant explorer quelques diagrammes UML pour illustrer notre système

I. Diagramme de cas d'utilisation



Description :

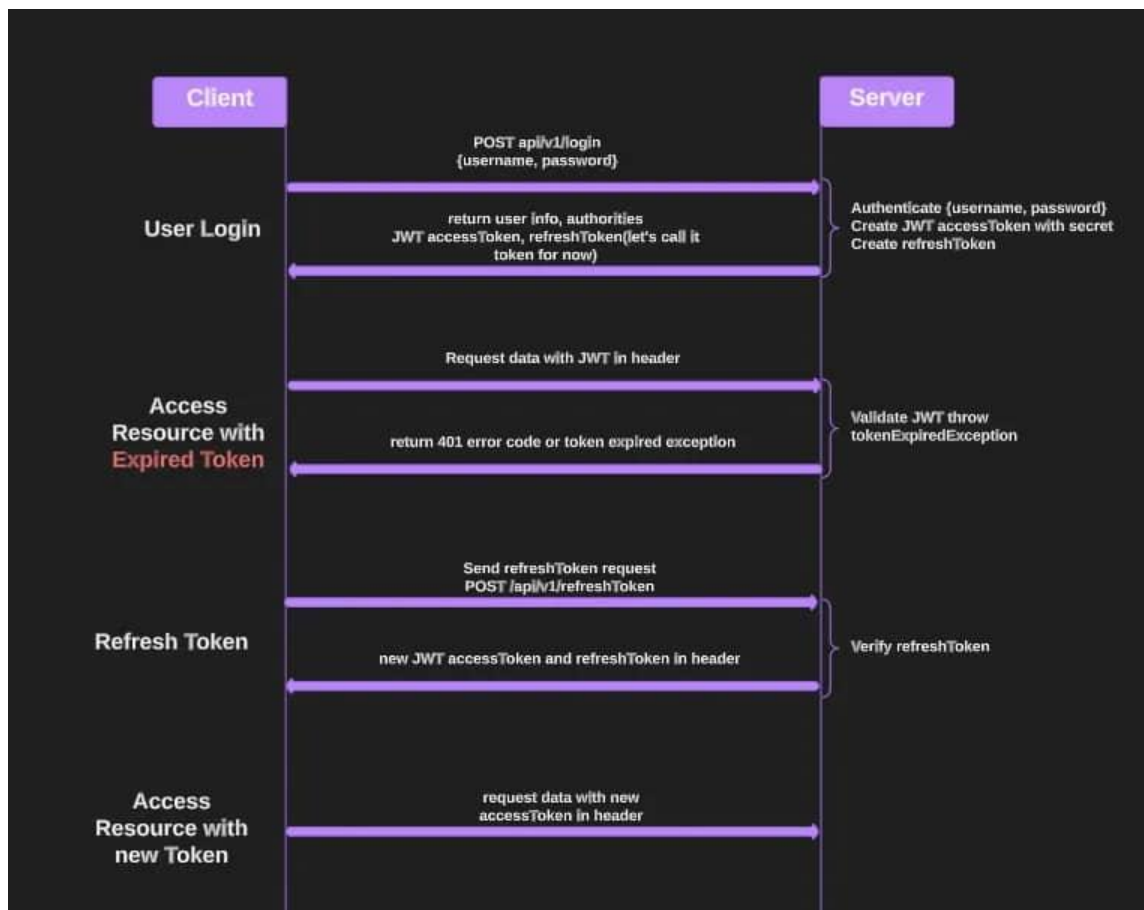
- L'utilisateur a la possibilité de s'authentifier avant d'utiliser notre application, charger une image pour extraire les données textes, charger un code-barres ou QR qui vont conduire vers un lien Web, Afficher un texte, ouvrir une application, etc. Pour qu'il puisse consulter l'historique de ses

traitements, il doit s'authentifier s'il a un compte. S'il n'a pas de compte il peut se connecter avec son Gmail ou créer un compte sur notre application.

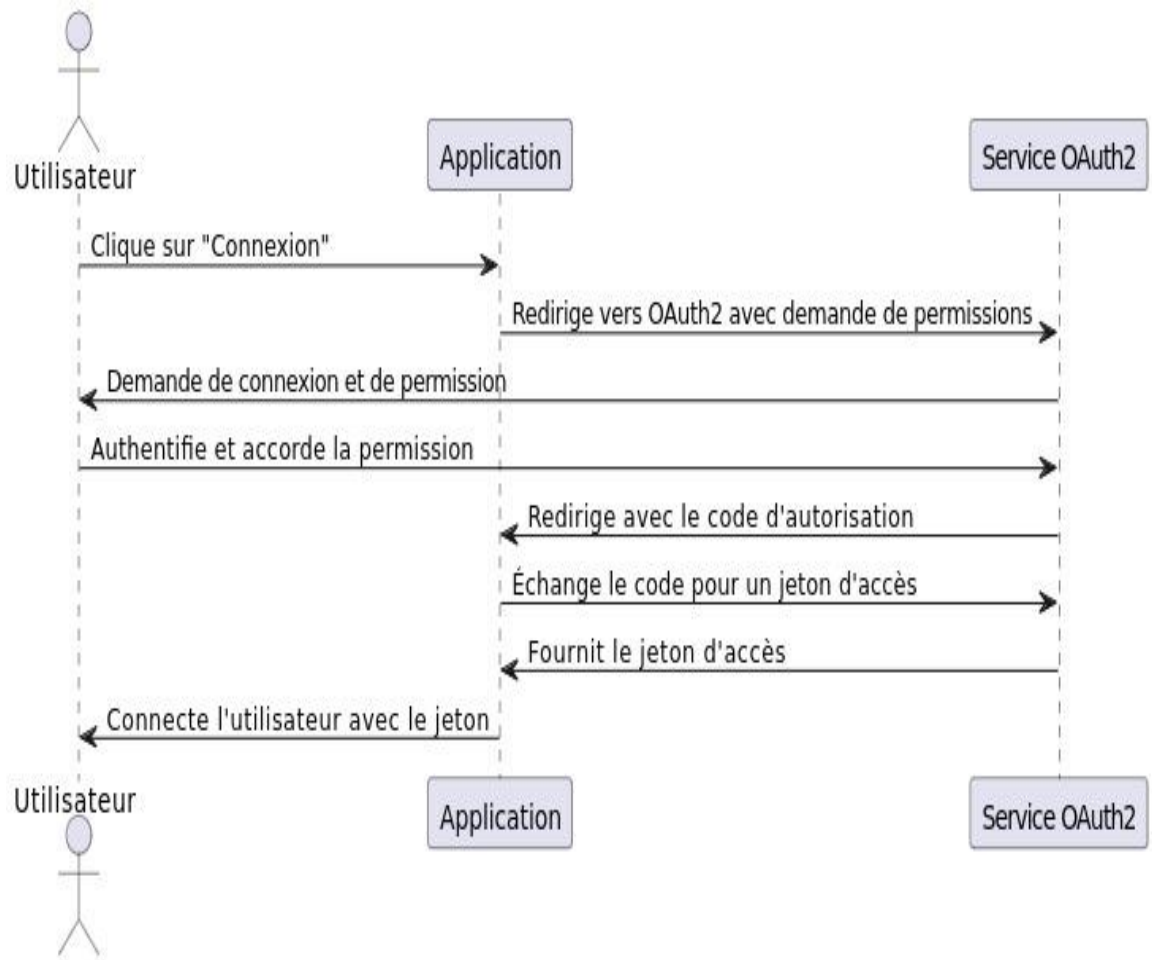
- L'administrateur a le droit de voir l'historique de traitement de tous les utilisateurs connectés ou pas. A aussi les mêmes cas d'utilisation que l'utilisateur.

IV. Diagramme de séquences

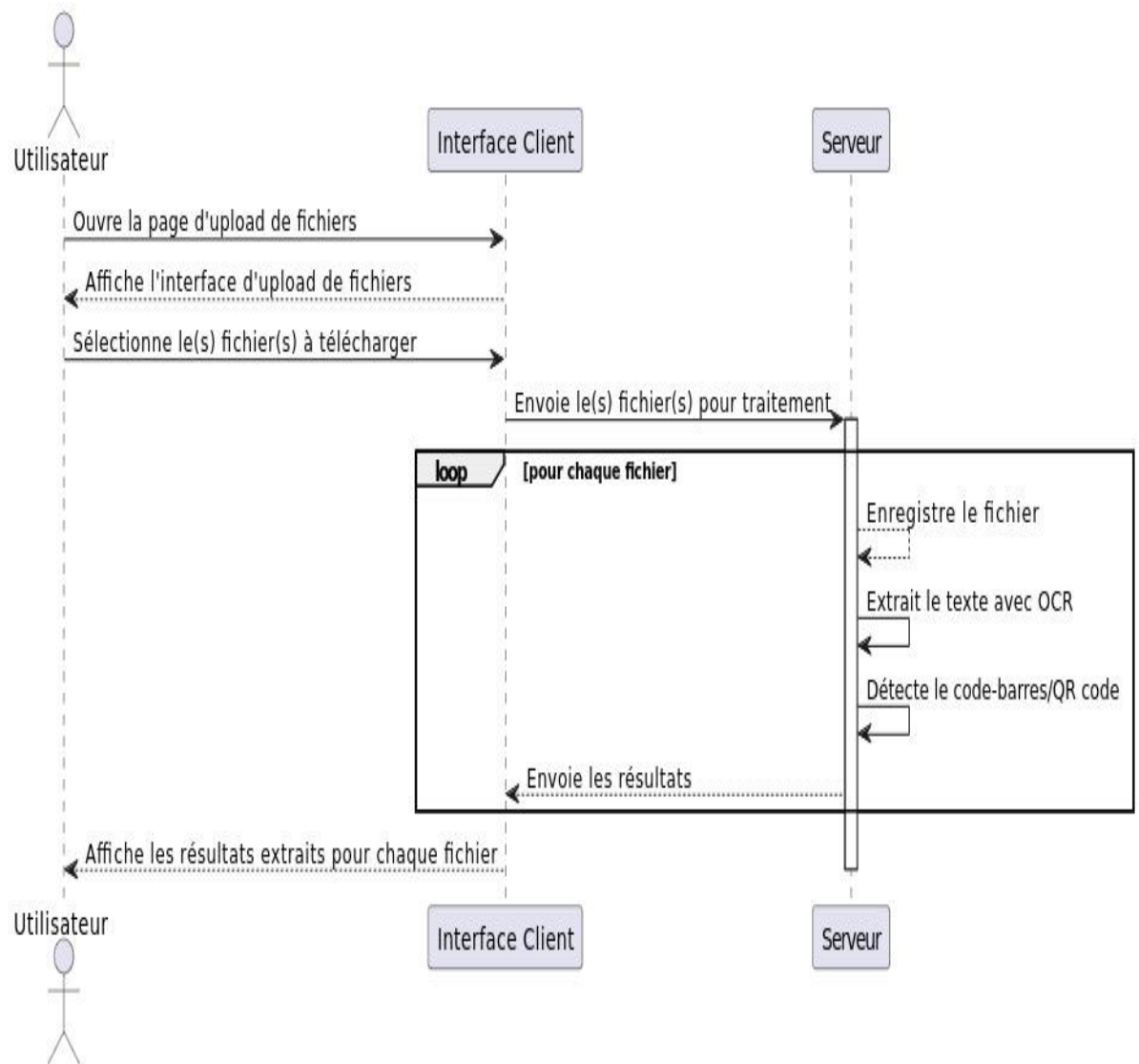
1- Diagramme de séquence Se connecter



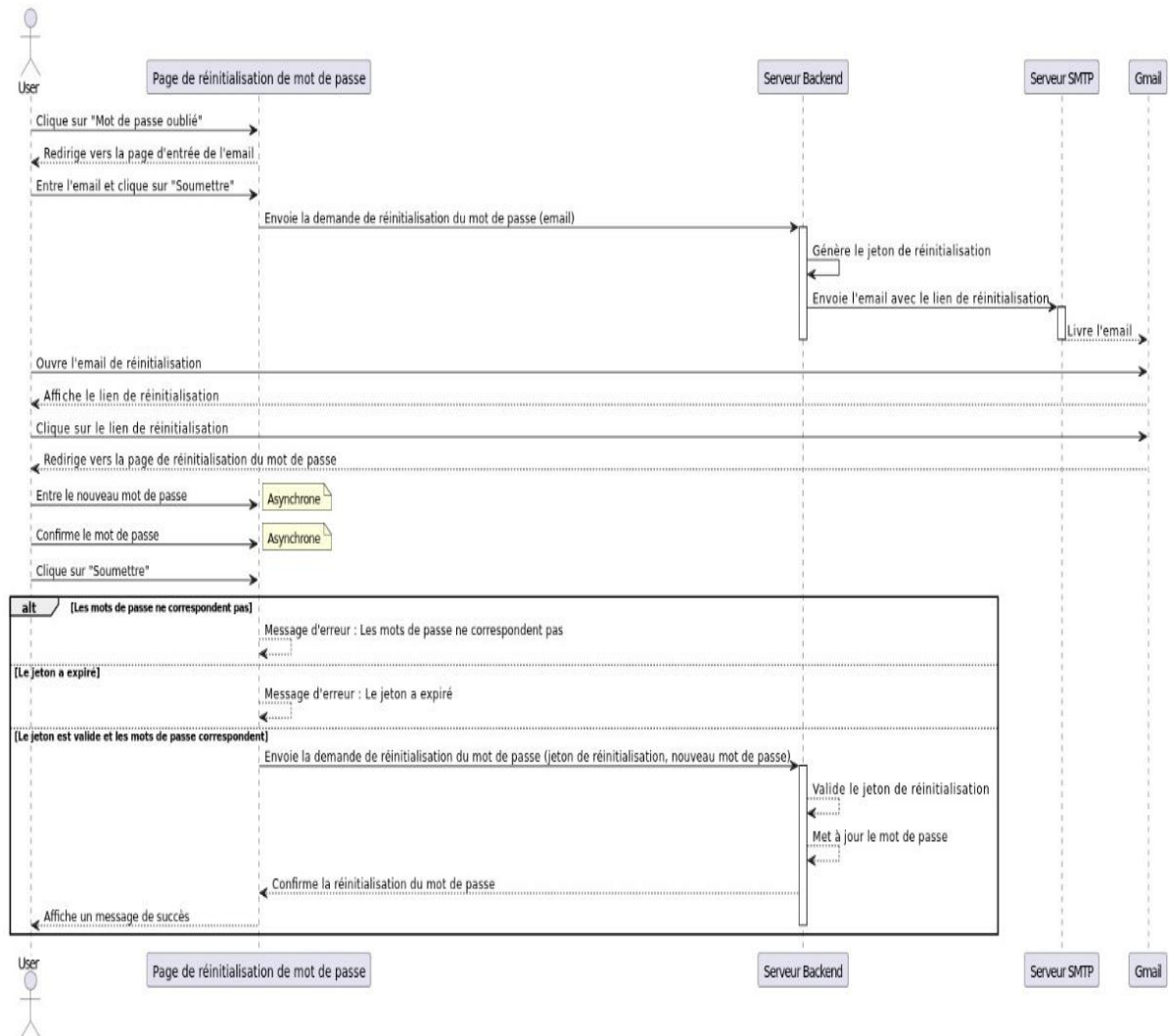
2- Diagramme de séquence Se connecter avec Gmail



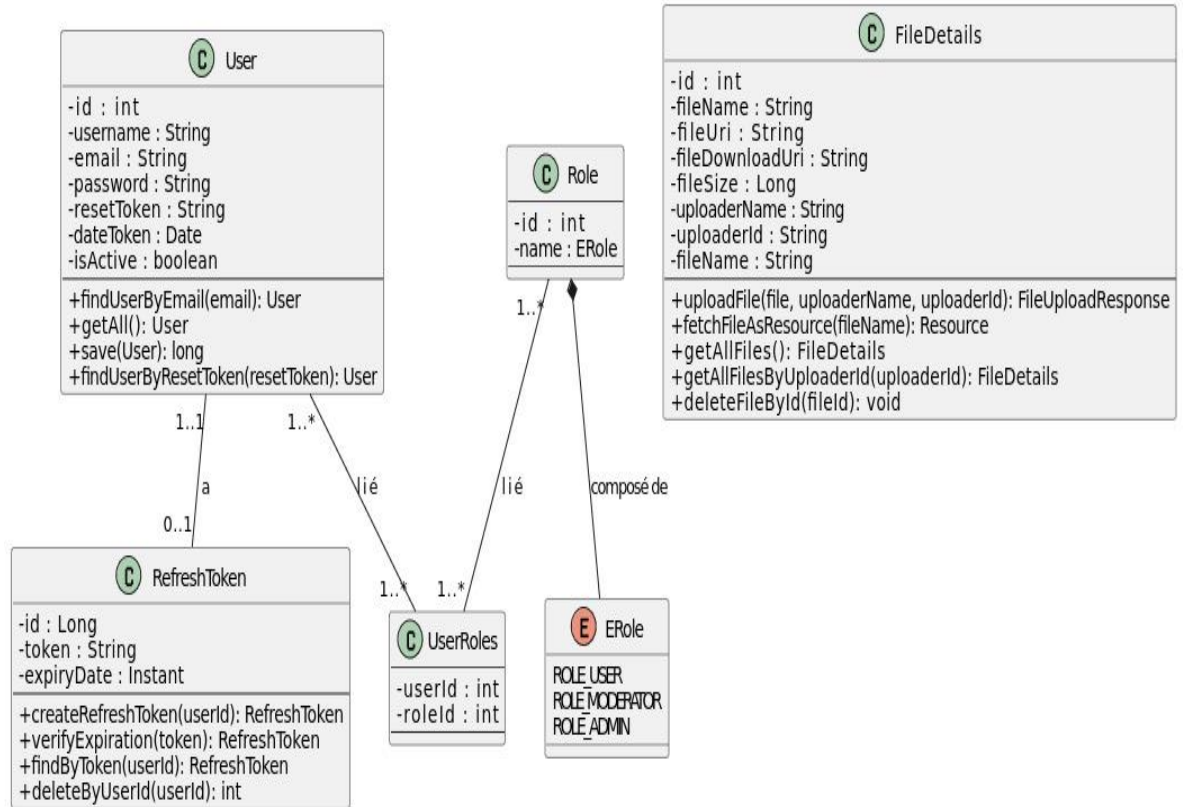
3- Diagramme de séquence upload de fichiers



4- Diagramme de séquence récupérer le mot de passe oublier pour l'utilisateur



V. Diagramme de classes



CHAPITRE 2 : IMPLEMENTATION TECHNIQUE

I. Environnements et technologies de développement

L'environnement de développement décrit utilise React.js pour le front-end, Spring JWT pour le backend, MySQL comme base de données, IntelliJ IDEA Community comme environnement de développement intégré (IDE), et Git comme outil de gestion de version. Cette combinaison de technologies permet de créer des applications web modernes et sécurisées, en utilisant des Framework et des outils bien établis dans l'industrie du développement logiciel.

II. Présentation de l'application



Salvador Profile

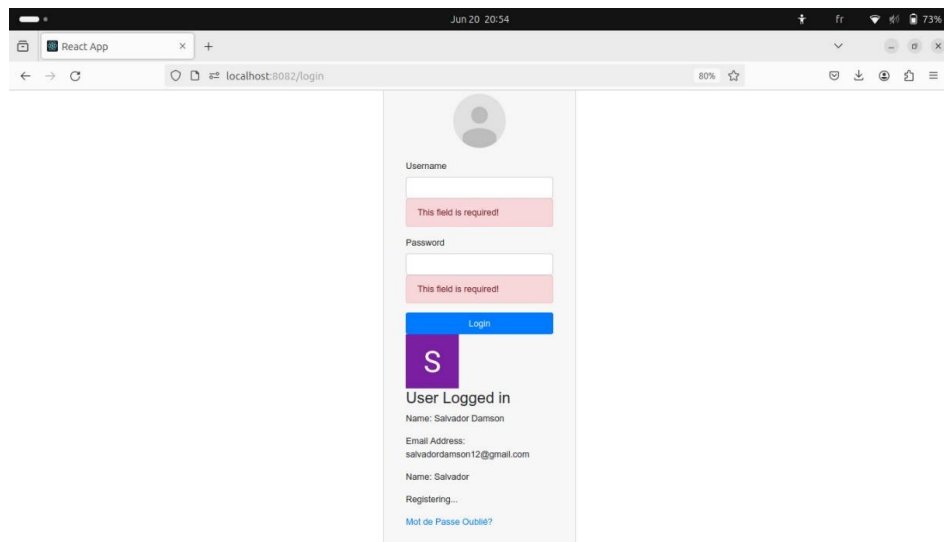
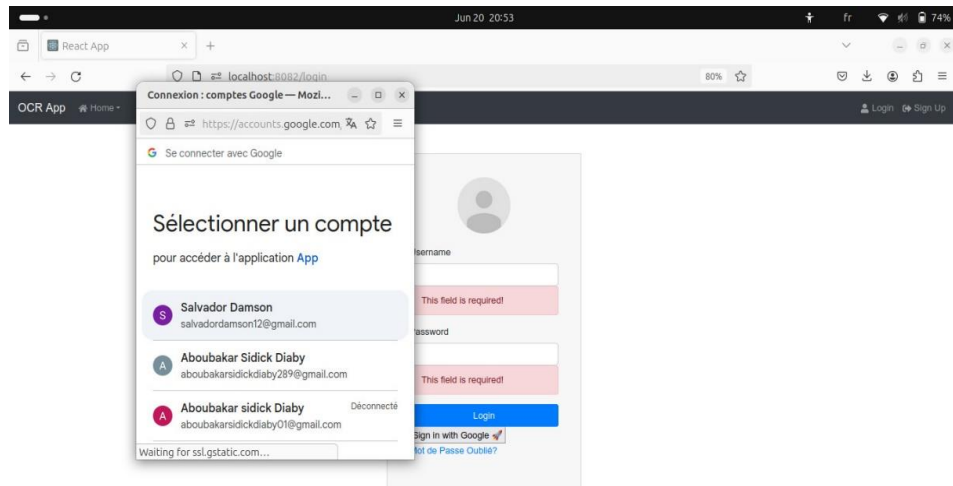
Token: eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXLTJ0IiwiaWF0Ij0iMTY1MjM0MjM0IiwiaXNjaWkiOiJ1PjR7XmRrQjAcerAI3Uj

Id: 6

Email: salvadordamson12@gmail.com

Authorities:

- ROLE_USER



Upload Image

Select a file:
 No files selected.

Uploader Name:



Upload Image

Select a file:
 6 files selected.

Uploader Name:



Upload Image

Select a file:
 6 files selected.

Uploader Name:



Text Extracted by OCR - 12201915052024_Screenshot from 2024-03-11 11-38-54.png

React App - Google Chrome 56 @ yB React App.x + << CG © localhost:3000/ledEmployee/18 aw Os Employee Management System
Made By Aboubakar Update Employee First Name Last Name Mabe Email marci2@gmail.com =>

Text Extracted by OCR - 55362305062024_Screenshot%20from%202024-06-05%2018-24-04.png

D 1 V3 Diaby-Aboubakar-sidick-FlowCV-Cover-Letter-20240531.pdf 172.8% - Q = - a x : DE) Diaby Aboubakar Sidick &
aboubakarsidickdiaby0 1@gmail.com Re 0501531152 Q Abidjan, Cote d'Ivoire ©) https://github.com/AboubakarRep Abidjan, le 31 mai
2024 Adressé au Directeur Général Des Ressources Humaines de la SNDI SNDI Abidjan - Plateau Boulevard Angoulvant Cité Financière
(Tour B) 4ème Etage Objet: Candidature spontanée stage soutenance developpeur fullstack B= © Monsieur/Madame,

Text Extracted by OCR - 22351107062024_th-1127541626.jpeg

j=

Data Extracted - 22351107062024_th-1127541626.jpeg

http://qr.rs/v/675a0

III. Défis et limites de l'implémentation

- **Défis techniques**

Bien que l'utilisation de Tesseract.js et Barcode Detector ait permis de résoudre une grande partie de nos problèmes, nous avons rencontré des défis liés à la fiabilité et à la précision de ces outils, en particulier avec des images de mauvaise qualité.

- **Limites de l'implémentation**

- **Sécurité et confidentialité :**

La manipulation des données des utilisateurs, notamment les informations contenues dans les codes-barres et QR codes, soulève des questions de sécurité et de confidentialité que nous devons prendre en compte.

- **Performances :**

Le traitement des images et l'extraction des données peuvent être gourmands en ressources, ce qui impacte les performances de l'application. Nous devons optimiser ces processus pour offrir une expérience fluide aux utilisateurs.

CONCLUSION GENERALE

Nous avons démontré avec succès notre capacité à développer une application web innovante qui extrait efficacement le texte, les codes-barres et les codes QR à partir d'images téléchargées par les utilisateurs. Malgré quelques défis techniques, notre solution offre des performances solides et une expérience intuitive pour les clients.

LA BIBLIOGRAPHIE, WEBOGRAPHIE

<https://medium.com/spring-boot/jwt-refresh-token-spring-security-c5b4646cddb9>

<https://www.descope.com/blog/post/oauth2-react-authentication-authorization>

https://developer.mozilla.org/en-US/docs/Web/API/Canvas_API/Tutorial/Drawing_shapes

<https://www.npmjs.com/package/tesseract.js/v/5.0.2>

<https://howtodoinjava.com/spring-boot/spring-boot-file-upload-with-react/>

<https://www.npmjs.com/package/barcode-detector>

<https://blog.logrocket.com/guide-adding-google-login-react-app/>

<https://github.com/abbes1960/Forget-and-Reset-password>

<https://github.com/abbes1960/Forget-And-Reset-Password-Partie-Front>

<https://docs.spring.io/spring-data/commons/docs/current/api/org/springframework/data/repository/CrudRepository.html>

<https://axios-http.com/docs/interceptors>

<https://dev.to/chafroudtarek/refresh-token-implementation-in-reactjs-53f7>

<https://devcamp.com/trails/comprehensive-react-development-tutorial/campsites/react-project-four-build-property-management-application/guides/how-to-store-auth-token-local-storage-react>

<https://permify.co/post/jwt-authentication-in-react/>

<https://clerk.com/blog/adding-jwt-authentication-to-react>