



HIGHER INSTITUTE OF SCIENCE (HIS)

RAPPORT TECHNIQUE

**Application Web de Gestion des
Réclamations SNTF**

Projet de Génie Logiciel

Réalisé par :

MEKHMOUKH RACHA

KIOUS LYLIA

OUCHENE MAROUA

BOUDJELTI BOUCHRA

BARR ROUMAISSA FARIEL

Enseignante :

Mme ZELLAGUI

Année Universitaire : 2024-2025

TABLE DES MATIÈRES

1	Introduction	4
1.1	Contexte du projet	4
1.2	Objectifs	4
2	Spécifications de l'Application Web	5
2.1	Acteurs du Système	5
2.1.1	Voyageur	5
2.1.2	Agent SNTF	5
2.1.3	Administrateur	5
2.2	Besoins Fonctionnels Principaux	6
2.2.1	Gestion des Réclamations	6
2.2.2	Interface Voyageur	6
2.2.3	Interface Agent	6
2.3	Besoins Non-Fonctionnels	6
2.3.1	Performance	6
2.3.2	Sécurité	6
2.3.3	Ergonomie	6
3	Diagramme d'activité	7
4	Diagramme des Cas d'Utilisation	9
4.1	Acteurs et Cas d'Utilisation Principaux	9
4.1.1	Voyageur	9
4.1.2	Agent SNTF	9
4.1.3	Administrateur	9
4.2	Relations entre Cas d'Utilisation	10
5	Processus Métier - Traitement d'une Réclamation	11

5.1	Flux Principal	11
5.1.1	Soumission par le Voyageur	11
5.1.2	Réception et Classification	11
5.1.3	Traitement par l'Agent	11
5.1.4	Résolution et Réponse	12
5.2	Flux Alternatifs	12
5.2.1	Escalade	12
5.2.2	Réclamation Incomplète	12
6	Diagramme de Classes	13
6.1	Classes Principales	13
6.1.1	Utilisateur	13
6.1.2	Voyageur (hérite de Utilisateur)	13
6.1.3	Agent (hérite de Utilisateur)	13
6.1.4	Reclamation	13
6.1.5	Reponse	13
6.2	Relations entre Classes	14
7	Diagramme d'États-Transitions	15
7.1	États d'une Réclamation	15
7.1.1	États Principaux	15
7.2	Transitions	15
7.2.1	De Nouvelle vers	15
7.2.2	De En Cours vers	15
7.2.3	De En Attente vers	16
7.2.4	De Résolue vers	16
8	Diagramme de séquence	17
8.1	Scénario : Soumission d'une réclamation	17
8.1.1	Acteurs et Composants	17
8.2	Diagramme de Séquence	18
9	Diagramme de conception	19
9.1	Architecture Logique Orientée Objet	19
9.1.1	Composants Clés	19
9.2	Relations UML	20
10	Diagramme de Déploiement	22
10.1	Architecture Technique	22
10.1.1	Serveur Web (Frontend)	22
10.1.2	Serveur d'Application (Backend)	22
10.1.3	Serveur de Base de Données	22
10.1.4	Services Externes	22
10.2	Protocoles de Communication	23
11	Recensement des classes	24
11.1	Utilisateur	24
11.2	Voyageur	24
11.3	AgentSNTF	24

11.4	Administrateur	25
11.5	Réclamation	25
11.6	Commentaire	25
11.7	TypeRéclamation	25
11.8	TableauDeBord	25
11.9	AuthenticationService	25
11.10	NotificationService	26
12	Outils de Développement	27
12.1	Environnement de Développement	27
12.1.1	IDE et Éditeurs	27
12.1.2	Gestionnaire de Versions	27
12.2	Technologies Frontend	27
12.2.1	Langages et outils	27
12.2.2	Éditeur & extensions	28
12.2.3	Styles et UI	28
12.3	Technologies Backend	28
12.3.1	Langage et Serveur	28
12.3.2	Base de Données	28
12.3.3	Sécurité	28
12.3.4	Documentation	28
12.4	Outils de Design et Prototypage	28
12.4.1	Maquettage	28
12.4.2	Diagrammes UML	28
12.4.3	Vidéo Explicative	29
13	Conclusion	30
13.1	Atouts de la solution proposée	30
13.2	Pistes d'amélioration	30
13.3	Impact attendu	31

CHAPITRE

1

INTRODUCTION

Dans le cadre de la modernisation des services publics en Algérie, la SNTF s'inscrit dans une dynamique de transformation numérique visant à renforcer la qualité de ses prestations. Ce rapport présente l'analyse, la conception et le développement d'une application web dédiée à la gestion des réclamations des voyageurs, en réponse aux besoins croissants de transparence, d'efficacité et de réactivité.

1.1 Contexte du projet

Face à la nécessité d'optimiser la relation client et de simplifier le processus de traitement des réclamations, la SNTF souhaite mettre en place une plateforme numérique accessible aux voyageurs. Cette solution offrira un canal de communication direct, fluide et structuré, permettant aux usagers de formuler leurs plaintes, de suivre leur avancement, et de recevoir des réponses dans les meilleurs délais.

L'application vise ainsi à moderniser l'interaction entre la SNTF et ses clients, tout en garantissant une meilleure traçabilité et un traitement rigoureux des réclamations.

1.2 Objectifs

- Faciliter la soumission de réclamations par les voyageurs
- Optimiser le traitement des réclamations par les agents SNTF
- Assurer un suivi transparent et en temps réel
- Améliorer la satisfaction client grâce à des réponses structurées

CHAPITRE

2

SPÉCIFICATIONS DE L'APPLICATION WEB

2.1 Acteurs du Système

2.1.1 Voyageur

- Créer et soumettre des réclamations
- Suivre l'état de ses réclamations
- Consulter l'historique de ses demandes
- Évaluer la qualité des réponses

2.1.2 Agent SNTF

- Consulter et traiter les réclamations assignées
- Modifier les statuts et ajouter des commentaires
- Accéder aux outils de tri et filtrage
- Utiliser le tableau de bord opérationnel

2.1.3 Administrateur

- Gérer les utilisateurs et leurs permissions
- Consulter les statistiques globales
- Configurer les paramètres du système
- Superviser les processus de traitement

2.2 Besoins Fonctionnels Principaux

2.2.1 Gestion des Réclamations

- Création avec catégorisation automatique
- Génération de numéro de suivi unique
- Système de notifications en temps réel
- Historique complet des interactions

2.2.2 Interface Voyageur

- Formulaire de réclamation intuitif
- Suivi en temps réel par numéro
- Espace personnel sécurisé
- Système d'évaluation de satisfaction

2.2.3 Interface Agent

- Dashboard avec vue d'ensemble
- Outils de filtrage et recherche avancée
- Gestion des priorités et escalade
- Système de commentaires internes

2.3 Besoins Non-Fonctionnels

2.3.1 Performance

- Temps de réponse ≤ 3 secondes
- Support de 100+ utilisateurs simultanés
- Disponibilité 99% (24h/24, 7j/7)

2.3.2 Sécurité

- Authentification JWT sécurisée
- Chiffrement des données sensibles
- Protection contre les attaques communes
- Logs d'audit complets

2.3.3 Ergonomie

- Interface responsive (mobile-first)
 - Conformité WCAG 2.0 niveau AA
 - Support multilingue (français/arabe)
 - Navigation intuitive
-

CHAPITRE

— 3 —

DIAGRAMME D'ACTIVITÉ

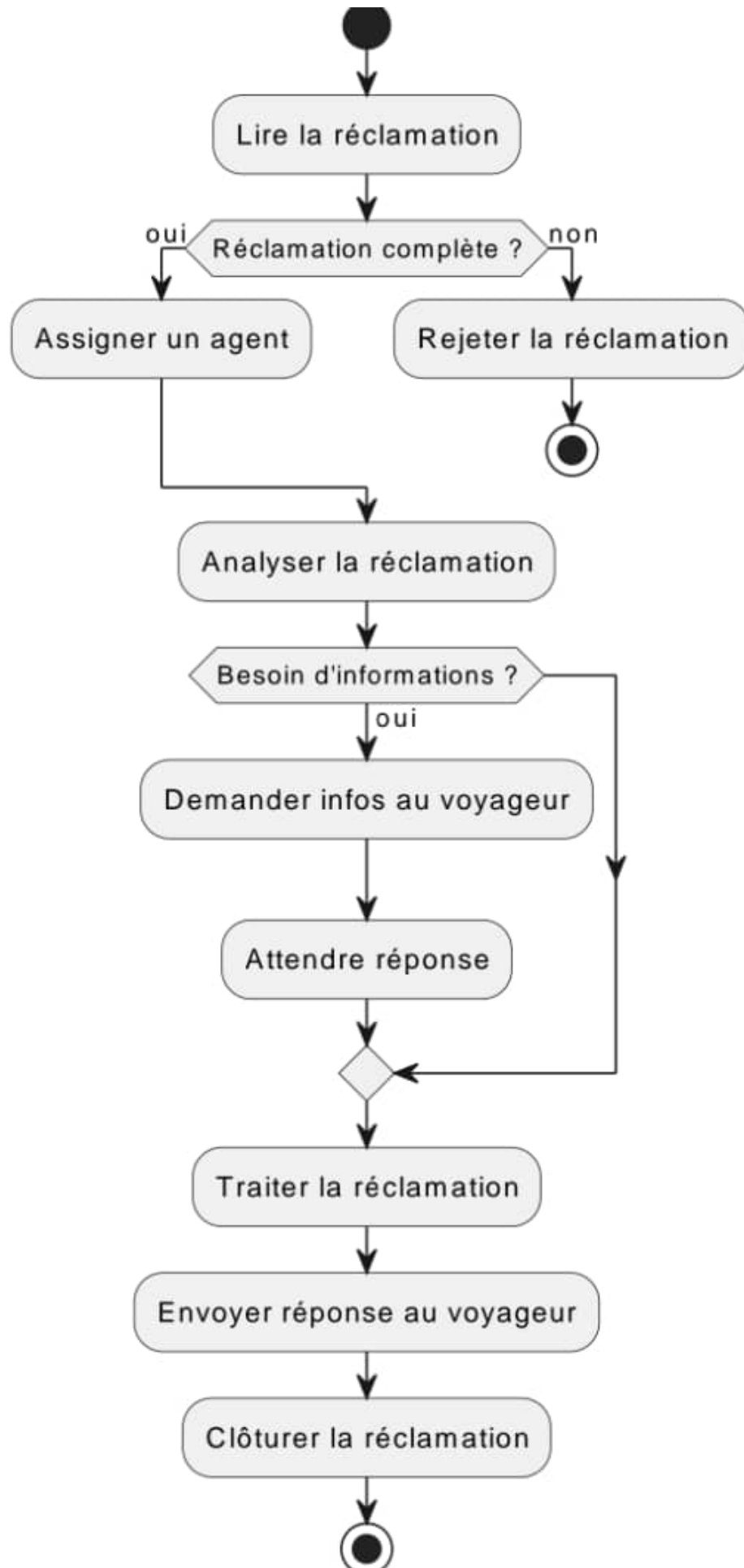


FIGURE 3.1 – Diagramme d'activité

CHAPITRE

4

DIAGRAMME DES CAS D'UTILISATION

4.1 Acteurs et Cas d'Utilisation Principaux

4.1.1 Voyageur

- S'inscrire/Se connecter
- Créer une réclamation
- Consulter ses réclamations
- Suivre une réclamation
- Évaluer une réponse

4.1.2 Agent SNTF

- Se connecter au système
- Consulter les réclamations assignées
- Traiter une réclamation
- Modifier le statut
- Rédiger une réponse
- Consulter le tableau de bord

4.1.3 Administrateur

- Gérer les utilisateurs
- Consulter les statistiques
- Configurer le système

- Gérer les catégories
- Superviser les agents

4.2 Relations entre Cas d'Utilisation

- **Inclusion** : "S'authentifier" inclus dans tous les cas nécessitant une connexion
- **Extension** : "Escalader vers superviseur" étend "Traiter réclamation"
- **Généralisation** : "Agent" et "Administrateur" héritent de "Utilisateur connecté"

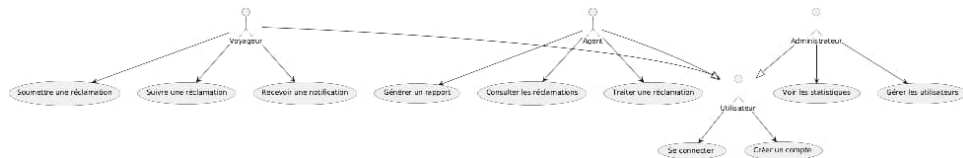


FIGURE 4.1 – Diagramme de cas d'utilisation

CHAPITRE

5

PROCESSUS MÉTIER - TRAITEMENT D'UNE RÉCLAMATION

5.1 Flux Principal

5.1.1 Soumission par le Voyageur

1. Saisie des informations de la réclamation
2. Sélection de la catégorie (retard, personnel, propreté, équipement)
3. Ajout éventuel de pièces jointes
4. Génération automatique du numéro de suivi

5.1.2 Réception et Classification

1. Enregistrement dans la base de données
2. Attribution automatique de priorité selon critères
3. Notification de confirmation au voyageur
4. Assignation à un agent selon la catégorie

5.1.3 Traitement par l'Agent

1. Consultation des détails de la réclamation
2. Investigation et analyse du problème
3. Changement de statut vers "En cours"
4. Communication interne si nécessaire

5.1.4 Résolution et Réponse

1. Élaboration de la réponse au voyageur
2. Mise à jour du statut vers "Résolue"
3. Notification automatique au voyageur
4. Archivage de la réclamation

5.2 Flux Alternatifs

5.2.1 Escalade

- Si délai dépassé ou complexité élevée
- Transfert vers superviseur
- Notification des parties concernées

5.2.2 Réclamation Incomplète

- Retour au voyageur pour complément
 - Statut "En attente d'information"
 - Relance automatique après délai
-

CHAPITRE

6

DIAGRAMME DE CLASSES

6.1 Classes Principales

6.1.1 Utilisateur

- **Attributs** : id, email, motDePasse, nom, prenom, telephone, role, dateCreation
- **Méthodes** : seConnecter(), seDeconnecter(), modifierProfil()

6.1.2 Voyageur (hérite de Utilisateur)

- **Attributs** : numeroClient
- **Méthodes** : creerReclamation(), consulterReclamations(), suivreReclamation()

6.1.3 Agent (hérite de Utilisateur)

- **Attributs** : matricule, service
- **Méthodes** : traiterReclamation(), modifierStatut(), redigerReponse()

6.1.4 Reclamation

- **Attributs** : id, numeroSuivi, titre, description, categorie, statut, priorite, dateCreation, dateModification
- **Méthodes** : genererNumeroSuivi(), changerStatut(), ajouterCommentaire()

6.1.5 Reponse

- **Attributs** : id, contenu, dateReponse, agentId

- Méthodes : rediger(), modifier(), publier()

6.2 Relations entre Classes

- Association : Voyageur (1) — (*) Reclamation
- Association : Agent (1) — (*) Reclamation
- Composition : Reclamation (1) — (*) Reponse
- Héritage : Utilisateur < | — Voyageur, Agent, Administrateur

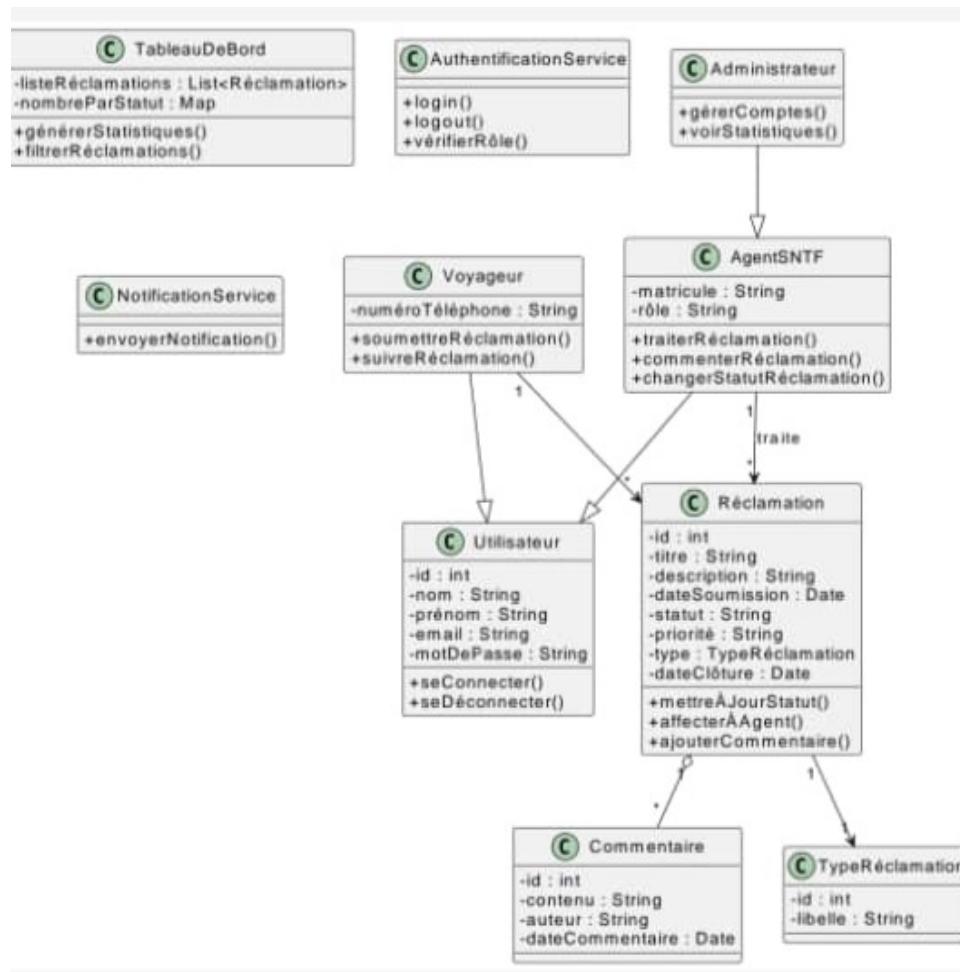


FIGURE 6.1 – Diagramme de classes

CHAPITRE

7

DIAGRAMME D'ÉTATS-TRANSITIONS

7.1 États d'une Réclamation

7.1.1 États Principaux

- **Nouvelle** : État initial après soumission
- **En Cours** : Réclamation prise en charge par un agent
- **En Attente** : Attente d'informations complémentaires
- **Résolue** : Traitement terminé avec réponse
- **Fermée** : Réclamation archivée après confirmation
- **Annulée** : Réclamation annulée par le voyageur

7.2 Transitions

7.2.1 De Nouvelle vers

- En Cours : prise en charge par agent
- Annulée : annulation par voyageur

7.2.2 De En Cours vers

- En Attente : demande d'informations complémentaires
- Résolue : résolution du problème
- Escaladée : transfert vers superviseur

7.2.3 De En Attente vers

- En Cours : réception des informations
- Fermée : timeout sans réponse

7.2.4 De Résolue vers

- Fermée : validation par voyageur
- En Cours : contestation de la réponse

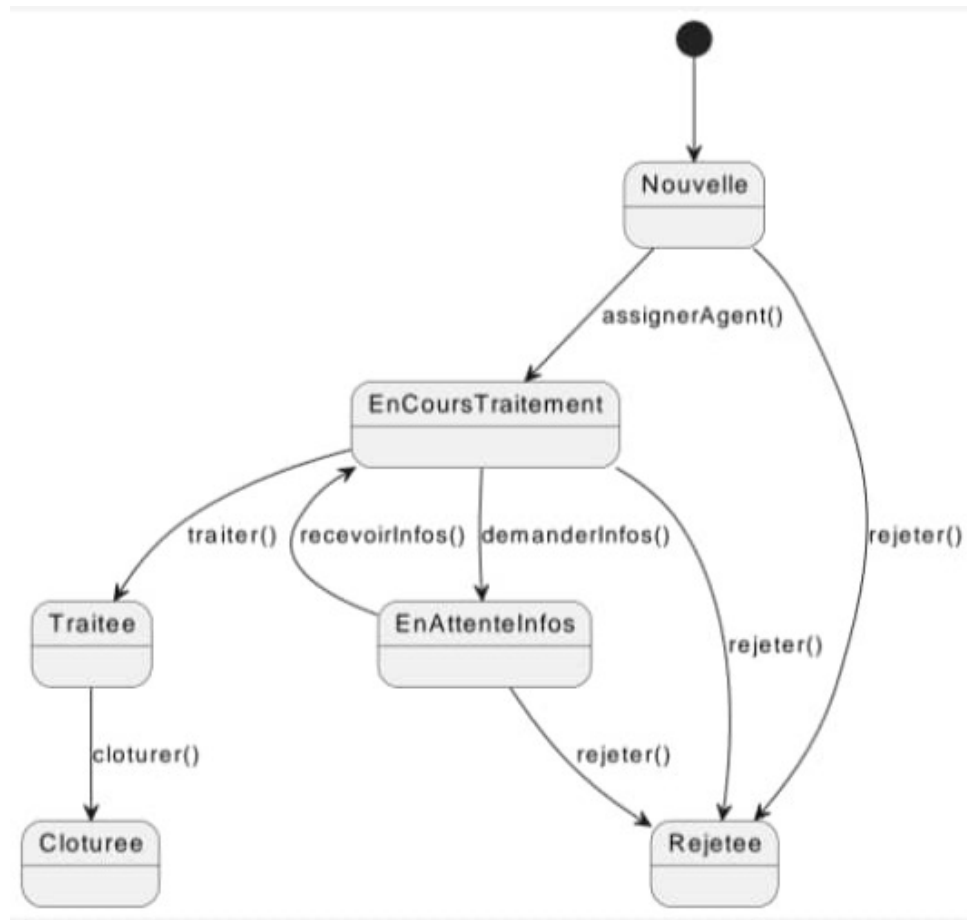


FIGURE 7.1 – diagramme d'état de transition

CHAPITRE

8

DIAGRAMME DE SÉQUENCE

8.1 Scénario : Soumission d'une réclamation

Ce diagramme illustre l'interaction entre les différents composants de l'application lors de la soumission d'une réclamation par un voyageur.

8.1.1 Acteurs et Composants

Voyageur (Frontend React.js)

- Remplit le formulaire de réclamation
- Soumet les données via l'interface

Interface Web

- Valide les champs du formulaire
- Appelle l'API REST

Backend (Express.js / Node.js)

- Reçoit la requête HTTP POST
- Vérifie l'authentification (JWT)
- Valide et insère la réclamation

Base de Données (PostgreSQL)

- Stocke la réclamation et ses métadonnées

- Gère les index et la cohérence

Service de Notification

- Envoie un email ou SMS de confirmation au voyageur
- Log l'événement dans le système de suivi

8.2 Diagramme de Séquence

Ce diagramme met en évidence la chronologie suivante :

1. Voyageur → Interface Web : remplirFormulaire()
2. Interface Web → Backend : envoyerReclamation()
3. Backend → DB : insertReclamation()
4. DB → Backend : OK
5. Backend → Service Notification : sendMail()
6. Notification → Voyageur : confirmation()
7. Backend → Interface Web : afficherMessage()

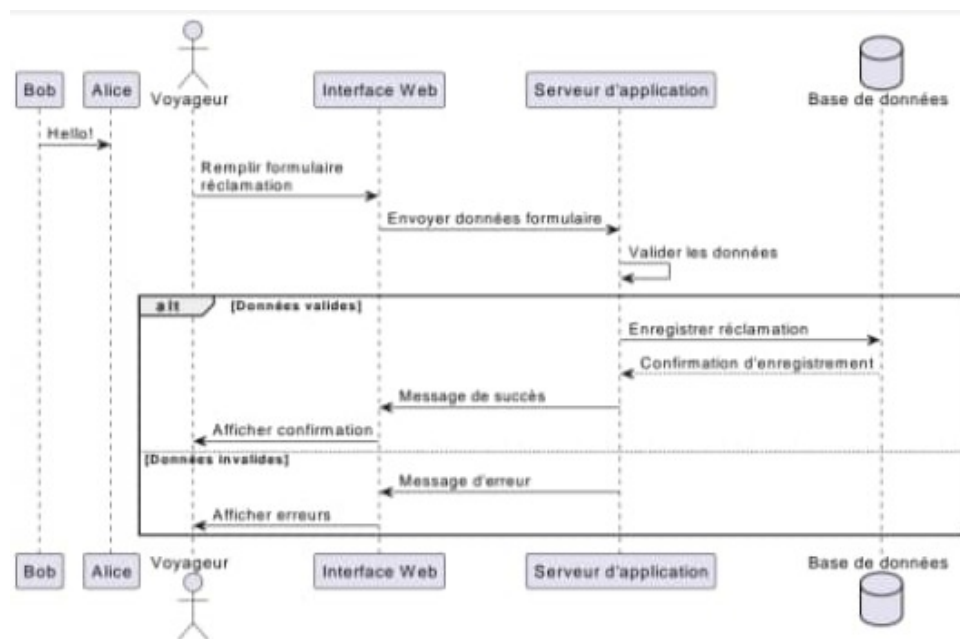


FIGURE 8.1 – diagramme de séquence

CHAPITRE

9

DIAGRAMME DE CONCEPTION

9.1 Architecture Logique Orientée Objet

Ce diagramme présente les principales classes métiers ainsi que leurs relations dans le cadre de la conception orientée objet de l'application.

9.1.1 Composants Clés

Classe Voyageur

- **Attributs** : id, nom, email, motDePasse
- **Méthodes** : seConnecter(), soumettreRéclamation(), consulterRéclamations()

Classe Réclamation

- **Attributs** : id, type, description, dateSoumission, statut, pièceJointe
- **Méthodes** : modifierStatut(), ajouterCommentaire()

Classe Agent

- **Attributs** : id, nom, email
- **Méthodes** : traiterRéclamation(), ajouterCommentaire()

Classe Commentaire

- **Attributs** : id, contenu, dateCommentaire

Classe Administrateur

- **Attributs** : id, nom, email
- **Méthodes** : gérerUtilisateurs(), consulterStatistiques()

Classe Notification

- **Attributs** : message, dateNotification
- **Méthode** : envoyer()

Énumération Statut

EN_ATTENTE, EN_COURS, RESOLUE, REJETEE, CLOTUREE

9.2 Relations UML

- Voyageur → 0..* Réclamation
 - Réclamation → 0..* Commentaire
 - Réclamation → Agent
 - Réclamation → Statut
 - Réclamation → 0..* Notification
 - Administrateur → Voyageur, Agent
-

Diagramme de conception - Application de gestion des réclamations

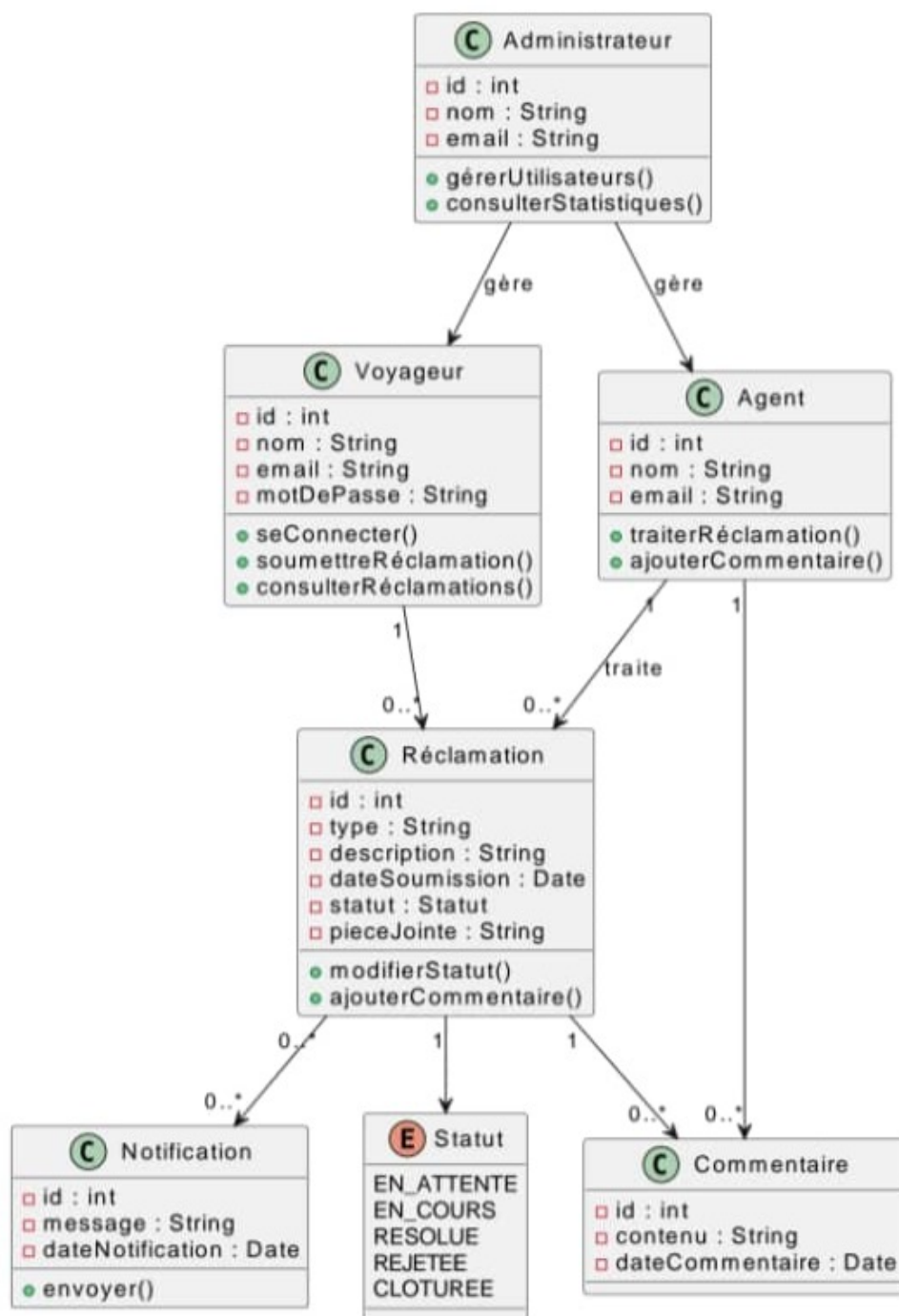


FIGURE 9.1 – diagramme de conception

CHAPITRE

10

DIAGRAMME DE DÉPLOIEMENT

10.1 Architecture Technique

10.1.1 Serveur Web (Frontend)

- React.js Application
- Serveur Nginx
- Certificat SSL/TLS
- CDN pour les assets statiques

10.1.2 Serveur d'Application (Backend)

- Node.js + Express.js
- API REST
- Middleware de sécurité
- Système d'authentification JWT

10.1.3 Serveur de Base de Données

- PostgreSQL 14+
- Système de sauvegarde automatique
- Réplication master-slave
- Monitoring des performances

10.1.4 Services Externes

- Service de notification (Email/SMS)

- Service de stockage de fichiers
- Service de logs centralisé
- Service de monitoring

10.2 Protocoles de Communication

- **HTTPS** : Communication client-serveur sécurisée
- **WebSocket** : Notifications temps réel
- **TCP/IP** : Communication base de données
- **SMTP** : Envoi d'emails
- **REST API** : Architecture de services

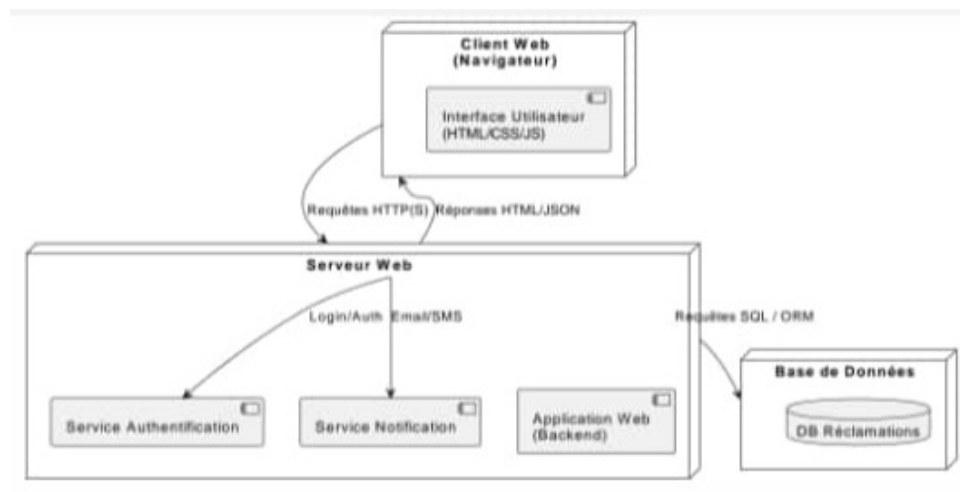


FIGURE 10.1 – diagramme de déploiement

CHAPITRE

11

RECENSEMENT DES CLASSES

11.1 Utilisateur

Classe abstraite représentant toute personne utilisant le système. Elle contient les informations de base pour l'authentification et la gestion des sessions.

Attributs : id, nom, prénom, email, motDePasse

Méthodes : seConnecter(), seDéconnecter()

11.2 Voyageur

Classe dérivée de Utilisateur. Elle représente les clients de la SNTF pouvant déposer et suivre des réclamations.

Attributs : numeroTéléphone

Méthodes : soumettreRéclamation(), suivreRéclamation()

11.3 AgentSNTF

Classe dérivée de Utilisateur. Elle représente les agents en charge du traitement des réclamations.

Attributs : matricule, rôle

Méthodes : traiterRéclamation(), commenterRéclamation(), changerStatutRéclamation()

11.4 Administrateur

Sous-classe d'AgentSNTE, représentant un utilisateur avec des droits supplémentaires d'administration.

Méthodes : gérerComptes(), voirStatistiques()

11.5 Réclamation

Classe principale du système, représentant une plainte déposée par un voyageur.

Attributs : id, titre, description, dateSoumission, statut, priorité, type, dateClôture

Méthodes : affecterÀAgent(), mettreÀJourStatut(), ajouterCommentaire()

11.6 Commentaire

Classe utilisée pour ajouter des remarques ou notes sur une réclamation. Chaque commentaire est lié à un auteur (généralement un agent).

Attributs : id, contenu, auteur, dateCommentaire

11.7 TypeRéclamation

Classe utilitaire permettant de classer les réclamations (par exemple : "retard", "propreté", "comportement du personnel", etc.).

Attributs : id, libelle

11.8 TableauDeBord

Classe permettant la visualisation et l'analyse des réclamations. Accessible par les agents et administrateurs.

Attributs : listeRéclamations, nombreParStatut

Méthodes : générerStatistiques(), filtrerRéclamations()

11.9 AuthentificationService

Classe chargée de gérer les connexions, déconnexions et la validation des rôles.

Méthodes : login(), logout(), vérifierRôle()

11.10 NotificationService

Service en charge de l'envoi de notifications aux utilisateurs (par exemple : confirmation de soumission, changement de statut).

Méthodes : envoyerNotification()

CHAPITRE

12

OUTILS DE DÉVELOPPEMENT

Note : Ce chapitre présente les technologies et outils utilisés pour le développement de l'application de gestion des réclamations SNTF.

12.1 Environnement de Développement

12.1.1 IDE et Éditeurs

- Visual Studio Code avec extensions
- Extensions : Prettier, GitLens, image preview, PHP, auto rename tag, live server, output colorizer
- Terminal intégré avec support Git

12.1.2 Gestionnaire de Versions

- Git intégré à VSCode avec deux extensions (GitLens, Git Graph)
- Repository GitHub
- Pushs réguliers pour suivre l'évolution

12.2 Technologies Frontend

12.2.1 Langages et outils

- HTML, CSS
- PHP côté frontend (pour l'affichage dynamique)

12.2.2 Éditeur & extensions

- Visual Studio Code avec extensions de base (Live Server, IntelliSense HTML/CSS, etc.)

12.2.3 Styles et UI

- Styling réalisé en CSS natif
- Responsive design géré manuellement avec media queries

12.3 Technologies Backend

12.3.1 Langage et Serveur

- PHP procédural/structuré manuellement
- Exécution via serveur local XAMPP (apache)

12.3.2 Base de Données

- MySQL
- Requêtes SQL écrites manuellement
- Base structurée avec phpMyAdmin

12.3.3 Sécurité

- Échappement des entrées utilisateur pour éviter les injections SQL
- Logique de sessions PHP pour l'authentification

12.3.4 Documentation

- Fichier README en place (présentation du projet, instructions de base)
- Documentation technique avec PhpDcumentor.

12.4 Outils de Design et Prototypage

12.4.1 Maquettage

- Figma pour prototypes interactifs
- Système de design cohérent
- Composants réutilisables

12.4.2 Diagrammes UML

- Diagrammes UML avec PlantText UML Editor
-

12.4.3 Vidéo Explicative

- Montage vidéo
- Enregistrement d'écran
- Sous-titres et voix-off

CHAPITRE

13

CONCLUSION

À l'issue de ce travail, nous avons conçu et spécifié une application web innovante permettant la gestion centralisée des réclamations des voyageurs de la SNTF. En partant des besoins fonctionnels identifiés jusqu'à la modélisation complète du système (UML, processus métier, base de données et interface), le projet propose une solution robuste, évolutive et orientée utilisateur.

13.1 Atouts de la solution proposée

- Une architecture claire et scalable basée sur des standards modernes.
- Une expérience utilisateur fluide, pensée pour la simplicité et l'accessibilité.
- Une sécurité renforcée pour la gestion des données personnelles.
- Une traçabilité complète des réclamations pour un meilleur suivi.
- Des outils adaptés aux agents pour un traitement efficace et rapide.

13.2 Pistes d'amélioration

- Connecter l'application aux systèmes internes de la SNTF (ex : gestion des horaires, billets).
- Déployer une version mobile native afin de renforcer l'accessibilité.
- Intégrer des fonctionnalités d'analyse intelligente (IA, NLP) pour prioriser les réclamations.
- Ajouter un espace partenaire ou API externe pour les agences de voyage.
- Mettre en place un module d'évaluation post-traitement par les voyageurs.

13.3 Impact attendu

- Valorisation de l'image de la SNTF comme acteur moderne du service public.
- Réduction du temps de réponse aux plaintes grâce à l'automatisation.
- Renforcement de la transparence dans la relation client.
- Amélioration continue via les statistiques et les retours utilisateurs.

En somme, cette application représente une avancée concrète vers une gestion numérique plus efficace et plus humaine des réclamations voyageurs, en parfaite adéquation avec les ambitions de transformation digitale de la SNTF.