



# Stage d'Initiation

Elève Ingénieur en 1ère année

Génie Informatique

Stage réalisé au sein de : Centre Hospitalier Universitaire  
Hassan II

**Application Web de Gestion des courriers du bureau  
d'ordre**

Période de stage : 02/06/2025 à 05/08/2025

**Réalisé par : Sabbab Youssef**

**Encadrant ENSAF :**

**Encadrant Société : Mme. Azzaoui Houda**

**Membres de jury :**

- M.
- M.
- M



## Remerciement

Je tiens à exprimer ma profonde gratitude à toutes les personnes qui ont contribué à la réalisation de ce projet de stage au sein du Centre Hospitalier Universitaire Hassan II de Fès.

En premier lieu, j'adresse mes sincères remerciements à Madame Houda AZZAOUÏ, mon encadrante au sein de la Division des Systèmes d'Information, pour son encadrement, ses précieux conseils et la confiance qu'elle m'a accordée tout au long de ce projet. Son expertise et son soutien constant ont été des éléments clés dans la réussite de cette expérience.

Je tiens également à remercier mon encadrant pédagogique à l'Ecole Nationale des Sciences Appliquées de Fès (ENSA Fès), Monsieur Z, pour ses orientations et sa disponibilité. J'adresse aussi mes remerciements à Monsieur X et Monsieur Y, membres du jury, pour leurs évaluations, leurs remarques pertinentes et leurs conseils éclairés.

Enfin, je suis reconnaissant envers toutes les personnes du CHU Hassan II et de l'ENSA Fès qui ont contribué de près ou de loin à la réussite de ce projet, ainsi qu'à ma famille pour leur soutien moral.



## Résumé

Ce rapport de stage d'initiation présente mon expérience au sein du Centre Hospitalier Universitaire Hassan II de Fès. Durant ce stage, j'ai travaillé sur le développement d'une application web pour la gestion des courriers de bureau d'ordre, visant à améliorer le processus de gestion des flux documentaires au sein du CHU. Le projet répond aux défis du système actuel de gestion des courriers, notamment le traitement manuel des courriers de type départ et arrivée, souvent source d'erreurs et de retards.

Le rapport détaille les objectifs principaux du projet : centralisation des informations, amélioration de la communication entre services hospitaliers, et suivi en temps réel des courriers. L'application développe des fonctionnalités de gestion des profils, d'enregistrement et de suivi des courriers. La modélisation UML avec diagrammes de classes et de cas d'utilisation a structuré l'architecture de l'application.

Le projet utilise des technologies modernes : React pour le front-end, Bootstrap pour la mise en forme, PHP pour le back-end, et MySQL pour la base de données, garantissant ainsi une solution robuste et scalable. Cette application centralisée améliore la coordination entre les différents services hospitaliers du CHU, réduit le temps de traitement des courriers et renforce la traçabilité et la transparence dans la gestion administrative hospitalière.

Ce stage a été une expérience enrichissante qui m'a permis d'appliquer mes connaissances académiques dans un cadre pratique, tout en développant des compétences en gestion de projet et en développement d'applications web, apportant une réelle valeur ajoutée à l'administration du CHU.



## Abstract

This internship report presents my experience at Hassan II University Hospital Center in Fez. During this internship, I worked on developing a web application for administrative mail management, aimed at improving the document flow management process within the CHU. The project addresses the challenges of the current mail management system, particularly the manual processing of incoming and outgoing mail, which is often a source of errors and delays.

The report details the main objectives of the project: centralization of information, improvement of communication between hospital services, and real-time mail tracking. The application develops profile management functionalities, mail registration and tracking. UML modeling with class diagrams and use case diagrams structured the application architecture.

The project uses modern technologies: React for the front-end, Bootstrap for styling, PHP for the back-end, and MySQL for the database, thus ensuring a robust and scalable solution. This centralized application improves coordination between the different hospital services of the CHU, reduces mail processing time, and strengthens traceability and transparency in hospital administrative management.

This internship was an enriching experience that allowed me to apply my academic knowledge in a practical framework, while developing skills in project management and web application development, bringing real added value to the CHU administration.



## Table des matières

Remerciement .....	1
Résumé .....	2
Abstract .....	3
Table des matières .....	4
Liste des figures.....	6
Introduction Générale .....	7
CHAPITRE I : Contexte général du projet .....	8
Introduction .....	9
1. Organisme d'accueil : .....	9
1.1 Présentation et Organigramme .....	9
1.2 Organigramme du CHU .....	10
1.3 Service Informatique .....	11
2. Présentation du projet .....	11
2.1 Contexte et Objectifs .....	11
2.2 Cahier des Charges .....	12
2.3 Conduite du projet .....	13
Conclusion .....	15
CHAPITRE II : Analyse et conception .....	16
Introduction .....	17
1. Analyse des besoins .....	17
1.1 Les besoins fonctionnels .....	17
1.2 Les besoins non fonctionnels .....	18
2. Méthodes d'analyse et conception .....	19
2.1 Le langage UML .....	19
3. Diagramme de cas d'utilisation .....	19
4. Diagramme de classe .....	20
Conclusion .....	21
CHAPITRE III : Mise en œuvre du projet .....	22
Introduction .....	23
1. Outils de développement .....	23
1.1 Outils Technologiques.....	23
1.2 Outils de base .....	24



1.3 Tests et Validation .....	25
2. Présentation de l'application .....	25
3. Le modèle MVC (Modèle-Vue-Contrôleur).....	38
4. API RESTful : Définition .....	38
Conclusion .....	38
Conclusion Générale .....	39
Bibliographie et Webographie .....	40



## Liste des figures

Figure 1 : logo de l'organisme d'accueil .....	10
Figure 2 : Organigramme de l'organisme d'accueil .....	10
Figure 3 : Organigramme de la division des systèmes d'information .....	11
Figure 4 : Le cycle de développement en Y .....	14
Figure 5 : Diagramme de Cas d'utilisation .....	20
Figure 6 : Diagramme de classes .....	21
Figure 7 Endpoint API de réception des courriers (courrier_recus) avec réponse JSON et en-têtes d'autorisation .....	25
Figure 8 : Informations invalides et message d'erreur .....	26
Figure 9 : Authentification tant qu'agent .....	27
Figure 10 : Interface d'utilisateur agent .....	27
Figure 11 : Choix de type de courrier .....	27
Figure 12 : Forme de formule de départ .....	28
Figure 13 : Message d'erreur si informations invalides .....	28
Figure 14 : Message de succès si courrier est diffusé .....	29
Figure 15 : Forme de formulaire de courrier d'arrivée .....	29
Figure 16 : Affichage de courriers valides .....	30
Figure 17 : Affichage de courrier valide .....	30
Figure 18 : Affichage de pdf de courrier .....	30
Figure 19 : Diffusion d'un courrier valide .....	31
Figure 20 : Affichage des courriers rejetés .....	31
Figure 21 : Affichage des courriers supprimés .....	31
Figure 22 : Affichage des courriers diffusés .....	32
Figure 23 : Button de Log out .....	32
Figure 24 : Connexion tant que responsable .....	32
Figure 25 : Interface responsable .....	32
Figure 26 : Affichage des courriers reçus .....	33
Figure 27 : Affichage d'un courrier reçu .....	33
Figure 28 : Validation du courrier reçu .....	34
Figure 29 : Rejection du courrier reçu .....	34
Figure 30 : Suppression du courrier reçu .....	35
Figure 31 : Affichage des courriers valides .....	35
Figure 32 : Connexion tant qu'Admin .....	35
Figure 33 : Interface Admin .....	36
Figure 34 : Statistiques 1 .....	36
Figure 35 : Statistiques 2 .....	36
Figure 36 : Création de user par succès .....	37
Figure 37 : Informations invalides et message d'erreur .....	37



## Introduction Générale

Dans le cadre du fonctionnement des établissements hospitaliers, la gestion des flux de courriers constitue un processus fondamental qui nécessite une organisation rigoureuse et une communication efficace entre les différents services. Les centres hospitaliers universitaires, confrontés à un volume important de correspondances administratives et médicales, utilisent encore des méthodes de gestion manuelles pour traiter les courriers entrants et sortants, rendant ce processus long, sujet à des erreurs, et parfois difficile à tracer.

C'est dans cette perspective que s'inscrit le projet développé lors de mon stage d'initiation au Centre Hospitalier Universitaire Hassan II de Fès. Ce projet consiste en la création d'une application web destinée à la gestion des courriers de bureau d'ordre. Cette application a pour objectif de centraliser les informations, d'automatiser les tâches administratives répétitives, et d'améliorer la communication entre les différents services hospitaliers et les gestionnaires du bureau d'ordre.

L'application développée vise à offrir une solution complète qui facilite l'enregistrement des courriers, leur classification par type (départ/arrivée), leur suivi en temps réel, et la génération de rapports statistiques. En s'appuyant sur des technologies modernes telles que React, Bootstrap, PHP, et MySQL, cette plateforme permet de réduire les délais de traitement et d'améliorer la précision des données.

Ce rapport est structuré en trois parties principales. La première partie présente l'organisme d'accueil et définit les objectifs de l'application. La deuxième partie aborde la phase de conception avec la méthodologie UML et les diagrammes de classes et de cas d'utilisation. La troisième partie détaille les outils de développement et présente les principales interfaces réalisées. Ce projet vise à offrir une solution robuste adaptée aux besoins du CHU, favorisant une meilleure coordination dans le traitement des courriers hospitaliers.





# CHAPITRE I :

## Contexte général du projet



## Introduction

Ce premier chapitre est fondamental pour ma compréhension du projet. Je commence par présenter l'organisme qui m'a accueilli, ce qui me permet de contextualiser mon stage. J'aborde ensuite la problématique du projet et son cadre d'intervention, en définissant clairement ses limites et objectifs.

Je poursuis en précisant les objectifs principaux de ma démarche, tout en expliquant la structure de ce document. Cette introduction me permet d'établir les bases nécessaires pour analyser les besoins, tant fonctionnels que techniques. Elle m'aide à élaborer un cahier des charges précis et à assurer une communication efficace avec les différentes parties prenantes, élément crucial pour la réussite du projet.

Je choisis de commencer logiquement par la présentation de l'organisme d'accueil, étape indispensable pour bien poser le cadre de mon étude.

## 1. Organisme d'accueil :

### 1.1. Présentation et Organigramme

Le Centre Hospitalier Universitaire (CHU) Hassan II de Fès, inauguré en 2009, est un établissement de référence au Maroc en matière de soins de santé. S'étendant sur 12 hectares avec une surface couverte de 78 102 m<sup>2</sup>, il dispose d'une capacité de 880 lits. Le CHU est composé de plusieurs hôpitaux spécialisés, notamment l'Hôpital des Spécialités, l'Hôpital Mère-Enfant, l'Hôpital d'Oncologie et de Médecine Nucléaire, l'Hôpital Omar Drissi, et l'Hôpital Ibn Al Hassan.

Ce centre hospitalier propose une large gamme de services médicaux, incluant les urgences, la réanimation, les soins intensifs, la chirurgie, et l'oncologie, tout en assurant la qualité et la continuité des soins. En plus de ses missions cliniques, le CHU Hassan II joue un rôle essentiel dans la recherche médicale et l'innovation, en particulier dans l'amélioration de la certification des causes de décès et la prévention des handicaps chez les enfants. Le CHU participe également à l'enseignement universitaire et postuniversitaire en médecine et pharmacie, ainsi qu'à la formation du personnel paramédical.



Figure 1: logo de l'organisme d'accueil

## 1.2. Organigramme du CHU

L'organigramme du Centre Hospitalier Universitaire Hassan II de Fès illustre une structure hiérarchique organisée sous la direction du Directeur Général, avec un Secrétaire Général supervisant plusieurs divisions stratégiques. Parmi celles-ci figurent la division des Affaires Financières (budget, comptabilité), la division des Ressources Humaines (gestion du personnel, formation), la division du Patrimoine et de la Maintenance (gestion des équipements et bâtiments), ainsi que la division des Systèmes d'Information (infrastructures, sécurité des systèmes, production statistique). En outre, des divisions dédiées à l'Enseignement et la Recherche et à l'Organisation des Affaires Médicales et Soins assurent la coordination des activités hospitalières. Chaque hôpital spécialisé, tels que l'Hôpital des Spécialités, l'Hôpital Mère-Enfant et l'Hôpital d'Oncologie, est dirigé de manière autonome, avec des services spécifiques dédiés aux finances, ressources humaines, soins infirmiers et logistique, garantissant une gestion efficace et adaptée à leurs besoins particuliers.



Figure 2: Organigramme de l'organisme d'accueil

### 1.3. Service Informatique

Le Service Informatique du Centre Hospitalier Universitaire Hassan II de Fès est structuré en plusieurs sous-sections spécialisées, regroupées sous la **Division des Systèmes d'Information**. Cette division se compose de plusieurs départements :

**Section des Infrastructures, Systèmes et Réseaux :** Responsable de la gestion et de la maintenance des infrastructures réseau, des systèmes informatiques et des connexions internes.

**Section de l'Exploitation des Systèmes d'Information:** Chargée de l'exploitation et de l'optimisation des systèmes d'information pour assurer leur bon fonctionnement au quotidien.

**Section de Sécurité des Systèmes d'Information :** Veille à la sécurité des données et à la protection contre les cybermenaces.

**Section de Production de l'Information et Statistiques:** Traite et analyse les données pour produire des rapports statistiques et des informations pertinentes à des fins de gestion et d'optimisation des services.

Cette organisation permet au service informatique de couvrir un large éventail d'activités nécessaires au bon fonctionnement des infrastructures technologiques de l'hôpital.

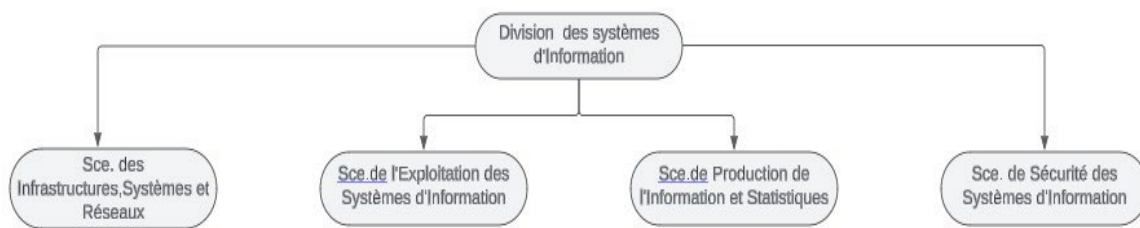


Figure 3: Organigramme de la division des systèmes d'information

## 2. Présentation du projet

### 2.1. Contexte et Objectifs

Le stage réalisé au Centre Hospitalier Universitaire Hassan II de Fès a consisté à développer une **Application Web de Gestion des Courriers du Bureau d'Ordre**. L'objectif de cette application est d'optimiser la gestion des courriers entrants et sortants de l'établissement, tout en garantissant une traçabilité complète et un suivi méticuleux de chaque document.

Dans le cadre de la mise en pratique des connaissances acquises en **Modélisation UML**, ce projet a visé à concevoir une solution centralisée pour la gestion documentaire. Cette solution intègre les différents acteurs organisationnels selon leurs niveaux de responsabilité et favorise une communication fluide entre les divers services de l'établissement.



## Objectifs Généraux de l'Application

- Faciliter la gestion centralisée des courriers au sein de l'organisation.
- Assurer un suivi rigoureux des courriers, qu'ils soient entrants ou sortants, en fonction de leur type (arrivée/départ).
- Offrir une interface utilisateur adaptée à chaque acteur, simplifiant le processus de traitement des courriers tout en garantissant une gestion fluide des interactions.
- Permettre une traçabilité complète des courriers et une meilleure coordination entre les différentes unités et services de l'organisation.

Ces objectifs permettent d'améliorer l'efficacité et la réactivité dans le traitement des courriers, tout en offrant une meilleure visibilité et transparence dans la gestion documentaire au sein du CHU.

## 2.2. Cahier des Charges

Préalablement au lancement du projet, nous avons établi un cahier des charges détaillé afin de préciser clairement les objectifs et les exigences fonctionnelles de l'application. Cette phase cruciale s'est déroulée au travers de multiples séances de travail avec mon encadrant, permettant d'identifier précisément les différents utilisateurs du système ainsi que les fonctionnalités spécifiques à leur proposer.

Cette approche méthodologique a assuré une compréhension approfondie des besoins organisationnels et a permis de définir un cadre de développement adapté aux réalités opérationnelles du Centre Hospitalier.

### Les acteurs:

Les principaux acteurs identifiés dans le système sont :

- Agents
- Responsables
- Administrateur

### Fonctionnalités par acteur

#### Agents :

Les agents ont accès aux fonctionnalités suivantes :

- Créer des courriers (de type départ ou arrivée).
- Envoyer des courriers validés.
- Consulter les courriers à valider (de type arrivée) en attente de validation par un responsable.
- Envoyer les courriers créés.



### *Responsables :*

Les responsables ont les fonctionnalités suivantes :

- Valider les courriers de type arrivée envoyés par les agents en ajoutant des annotations si nécessaire.
- Choisir l'unité, la division ou le service vers lequel le courrier doit être envoyé.
- Retourner les courriers avec des erreurs à l'agent avec un motif de rection.
- Supprimer les courriers obsolètes ou non conformes aux procédures établies.

### *Administrateur:*

Les administrateurs ont les fonctionnalités suivantes :

- Vérifier les utilisateurs : Consulter le nombre total d'utilisateurs et leur répartition par catégorie (agents, responsables, admins) ainsi le nombre des unités internes/externes.
- Consulter les statistiques : Vérifier le nombre de courriers (arrivée, départ, diffusés) et leur statut (valides, supprimés, rejetés).
- Consulter les statistiques : Vérifier le nombre de courriers (arrivée, départ, diffusés) et leur statut (valides, supprimés, rejetés).

### *Unités Internes et Externes*

- Unités internes : Composes de Sections , Divisions et Hôpitaux
- Unités externes : désignent l'ensemble des organismes, institutions et entités publiques ou privées qui sont extérieurs à l'établissement hospitalier mais qui entretiennent des relations administratives, techniques ou de coopération avec celui-ci dans le cadre de ses activités.

## **2.3. Conduite du projet**

Dans le cadre de notre projet, nous avons décidé d'adopter le modèle de développement en Y, qui repose sur deux dimensions principales : l'aspect fonctionnel et l'aspect technique .

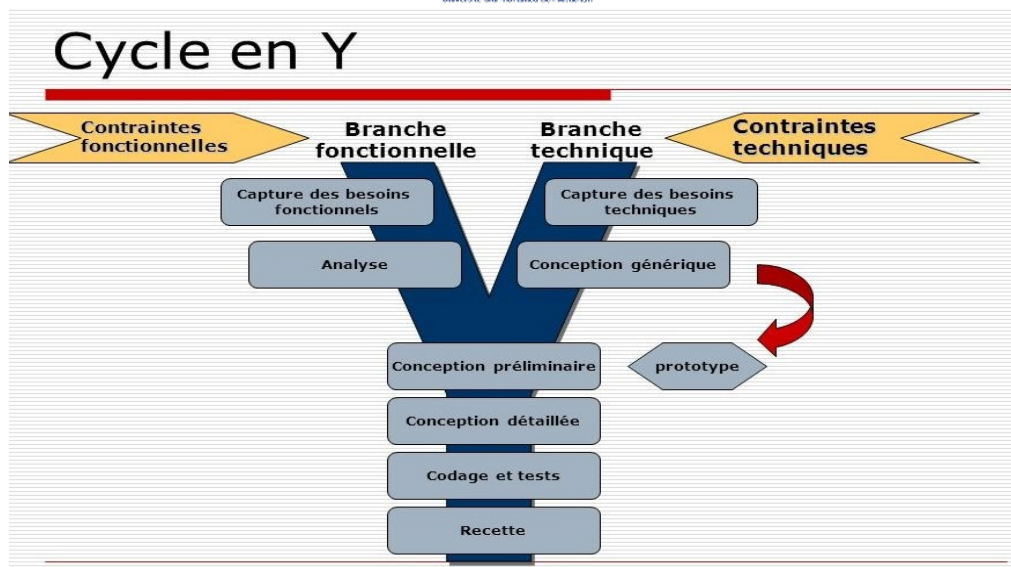


Figure 4: Le cycle de développement en Y

### Branche Fonctionnelle (branche gauche)

1. **Capture des Besoins Fonctionnels** : Cette phase génère un modèle des exigences orienté utilisateur. Elle permet d'évaluer rapidement le risque de développer un système inadéquat. La maîtrise d'œuvre procède à la consolidation des spécifications et en contrôle la cohérence ainsi que la complétude.
2. **Analyse** : Cette étape consiste à examiner minutieusement les spécifications fonctionnelles afin d'obtenir une compréhension précise des fonctionnalités que le système doit implémenter. Les conclusions de cette analyse demeurent indépendantes de toute solution technologique particulière.

### Branche Technique (branche droite)

1. **Capture des Besoins Techniques** : Cette phase inventorie l'ensemble des contraintes et des choix impactant la conception du système. Les outils et équipements retenus, ainsi que l'intégration aux systèmes existants, définissent habituellement les exigences techniques.
2. **Conception Générique** : Cette étape établit les composants requis pour élaborer l'architecture technique. Autonome vis-à-vis des aspects fonctionnels, elle vise à standardiser et réutiliser les mêmes mécanismes à travers l'ensemble du système. Un prototype est fréquemment préconisé pour valider cette conception et réduire les risques techniques.

### Branche Centrale

1. **Conception Préliminaire** : Cette phase intègre le modèle d'analyse dans l'architecture technique, établissant ainsi la cartographie des composants à développer.
2. **Conception Détaillée** : Cette étape consiste à définir les modalités de réalisation de chaque composant du système.



3. **Codage et Tests** : Au cours de cette phase, les composants sont développés et testés progressivement.
4. **Recette** : L'étape finale consiste à valider les fonctionnalités du système développé.

## Conclusion

En synthèse, ce premier chapitre a établi les bases méthodologiques de notre approche. Il s'est ouvert par la présentation de l'organisme d'accueil, permettant de contextualiser l'environnement professionnel du stage. Par la suite, la problématique, le périmètre du projet ainsi que ses objectifs ont été explicités de manière précise. Ce chapitre ambitionne essentiellement de préparer l'analyse détaillée des exigences projet, qu'elles soient fonctionnelles ou techniques, en perspective de la rédaction d'un cahier des charges rigoureux. Il forme ainsi l'assise méthodologique garantissant la réussite de notre démarche.





# CHAPITRE II :

## Analyse et conception



## 1. Analyse des besoins

### 1.1. Les besoins fonctionnels

Les besoins fonctionnels décrivent les fonctionnalités attendues de l'application pour répondre aux besoins des utilisateurs. Voici les principaux besoins fonctionnels identifiés pour l'application Gestion des Flux Bureau d'Ordre :

- **Gestion des courriers :**

Fonctionnalités pour les Agents

- **Création de courriers :** Les agents peuvent créer deux types de courriers :
  - Courriers de **départ** : diffusés automatiquement vers les unités internes ou externes
  - Courriers d'**arrivée** : transmis automatiquement au responsable pour validation
- **Visualisation des courriers rejetés :** Consultation des courriers refusés avec affichage détaillé des motifs de rejet pour correction
- **Suivi des courriers supprimés :** Accès à l'historique des courriers supprimés pour traçabilité
- **Consultation des courriers validés :** Visualisation des courriers approuvés par le responsable et diffusés dans le système

Fonctionnalités pour les Responsables

- **Réception des courriers d'arrivée :** Consultation des courriers créés par les agents nécessitant une validation
- **Processus de validation :** Possibilité de :
  - **Valider** le courrier pour diffusion
  - **Rejeter** le courrier avec indication du motif de refus
  - **Supprimer** le courrier du système
- **Suivi des courriers validés :** Accès à l'historique complet des courriers approuvés et diffusés par le responsable
- **Gestion centralisée :** Vue d'ensemble sur tous les courriers traités avec leur statut respectif (validé, rejeté, supprimé)



## Fonctionnalités pour l'administrateur

- **Statistiques et reporting :**

- Génération de statistiques sur les courriers par unité, division, et type de courrier (arrivée/départ).
- Consultation des statistiques par les administrateurs pour suivre la performance des unités.

### 1.2. Les besoins non fonctionnels

Les besoins non fonctionnels concernent la qualité du système et les contraintes techniques auxquelles l'application doit répondre. Ils sont essentiels pour garantir une utilisation efficace de l'application.

- **Performance :**

- L'application doit répondre rapidement aux requêtes des utilisateurs, même en cas d'utilisation intensive par plusieurs acteurs.
- Les temps de réponse doivent être optimisés pour garantir une expérience fluide.

- **Sécurité :**

- La gestion des accès doit être rigoureusement contrôlée avec des rôles bien définis pour chaque utilisateur.
- Les données des courriers doivent être protégées contre tout accès non autorisé par des mécanismes d'authentification.

- **Scalabilité :**

- L'application doit être capable de s'adapter à une augmentation du nombre d'utilisateurs et de courriers à gérer sans compromettre les performances.

- **Fiabilité et disponibilité :**

- L'application doit être disponible en tout temps.
- Les défaillances doivent être minimisées pour assurer une continuité du service.

- **Maintenance et évolutivité :**

- Le système doit être facile à maintenir et permettre l'ajout de nouvelles fonctionnalités ou modules sans perturber les utilisateurs.

- **Interface utilisateur :**

- L'interface doit être intuitive, ergonomique et adaptée à chaque rôle (administrateur, agent, responsable)
- Une attention particulière doit être accordée à l'expérience utilisateur pour garantir une prise en main rapide et efficace.



## 2. Méthodes d'analyse et conception

### 2.1. Le langage UML

Pour notre projet, nous avons choisi d'utiliser UML pour la modélisation orientée objet, une des caractéristiques clés du modèle de développement en Y.

#### • Justification du choix d'UML

- UML est un langage de modélisation reconnu qui facilite la communication entre les équipes et les parties prenantes.
- Il permet de représenter de manière claire les relations entre les classes, objets et leurs interactions.
- UML aide à formaliser et visualiser les exigences fonctionnelles et techniques, rendant le système plus compréhensible.
- Il permet d'anticiper et corriger les incohérences avant le développement, réduisant ainsi les erreurs.
- UML offre une documentation visuelle précise et complète, utile pour la maintenance et les mises à jour.
- Il facilite la modélisation des aspects fonctionnels et techniques, en garantissant leur cohérence lors de l'intégration.

## 3. Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation modélise un système de gestion de courrier électronique où chaque action requiert une authentification préalable. Les différents acteurs disposent de privilèges spécifiques : l'administrateur, qui gère les utilisateurs et les unités, ainsi que consulte les statistiques globales du système ; les agents, qui créent et envoient des courriers, et consultent également les statistiques concernant l'unité à laquelle ils appartiennent ; les responsables, chargés de valider, retourner les courriers, gérer les unités externes, et consulter les statistiques de leur unité ; et les chefs de service et de division, qui se limitent respectivement à l'envoi et au partage des courriers. La consultation des courriers est un droit accordé à plusieurs acteurs.

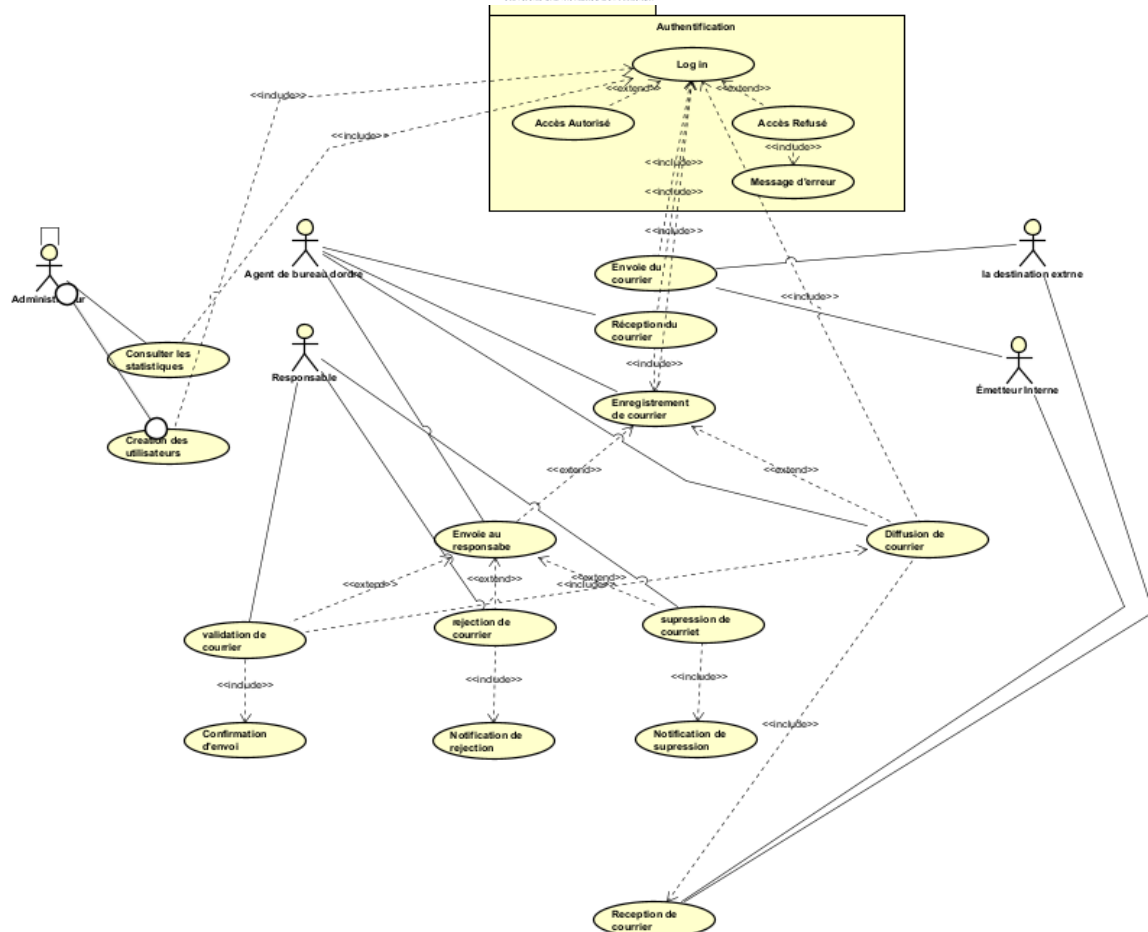


Figure 5: Diagramme de Cas d'utilisation

## 4. Diagramme de Classe

Le diagramme de classes représente la structure statique du système en définissant les classes et les relations entre elles. Il sert à modéliser les entités du système d'information en organisant les données en classes distinctes.

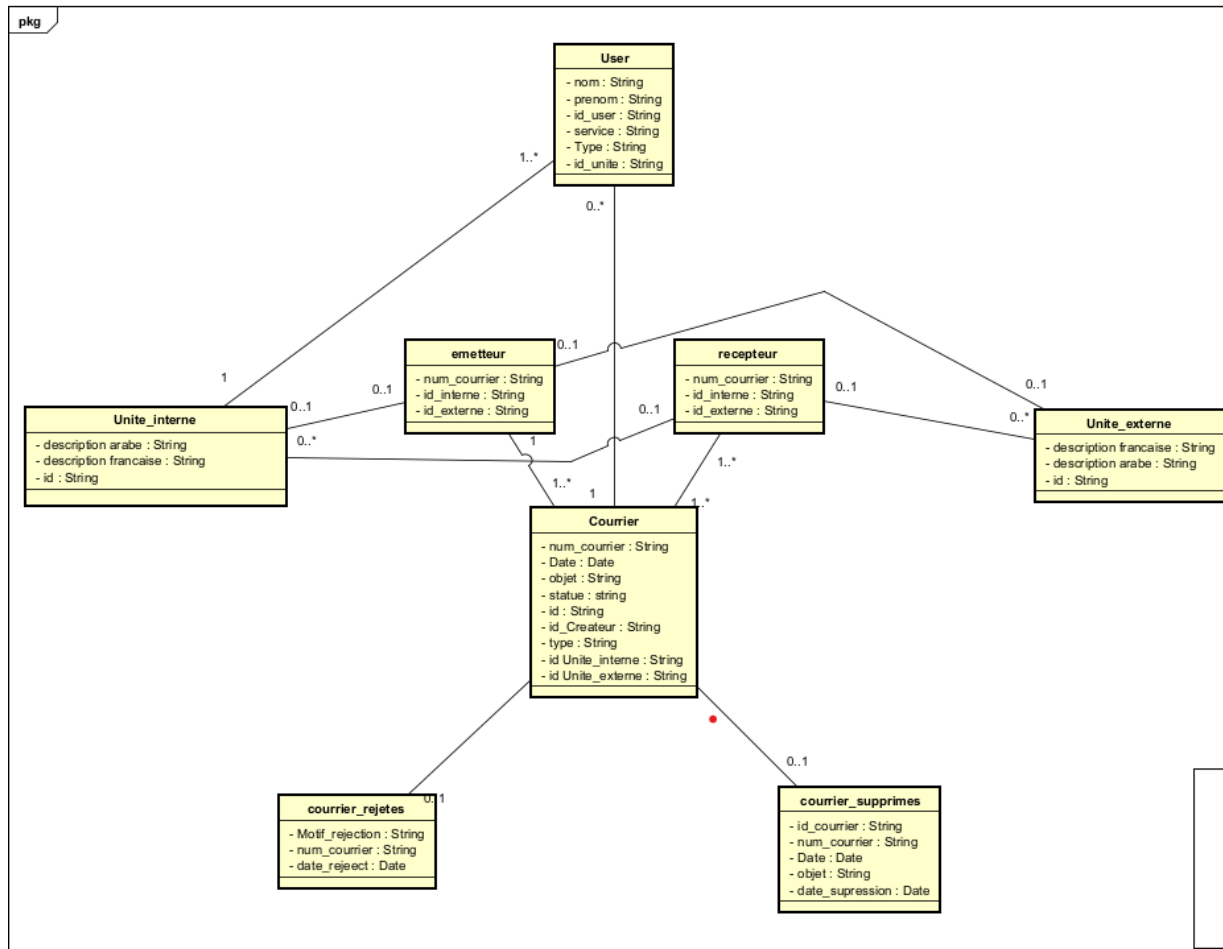


Figure 6: Diagramme de classes

## Conclusion

En conclusion, ce chapitre a permis de définir de manière approfondie les besoins fonctionnels et non fonctionnels de l'application web dédiée à la gestion du des Flux Bureau d'Ordre du CHU. L'analyse exhaustive des exigences a jeté les bases essentielles pour le développement à venir. La méthodologie UML a été justifiée comme un choix judicieux pour la modélisation, offrant une approche visuelle et documentée. Ces éléments constituent les fondements nécessaires à la création d'une plateforme robuste, répondant aux attentes des utilisateurs tout en garantissant des performances optimales et une sécurité accrue. Ce chapitre marque ainsi le point de départ concret vers la réalisation pratique de notre projet dans le contexte spécifique de CHU.



# CHAPITRE III :

## Mise en œuvre du projet



## Introduction

Dans le troisième chapitre dédié à la mise en œuvre de notre projet, nous nous plongerons au cœur du développement en explorant les choix stratégiques des outils et langages. Cette étape cruciale de notre parcours consolidera nos décisions technologiques, mettant en lumière les langages de programmation sélectionnés et les outils de base indispensables à la concrétisation de notre application. De plus, nous offrirons un aperçu des interfaces utilisateur, soulignant leur rôle central dans l'expérience immersive que nous cherchons à offrir aux utilisateurs. Par ailleurs, nous avons utilisé des outils pour la réalisation des tests unitaires, assurant la qualité et la robustesse de notre code, ainsi que des outils pour le déploiement de l'application, facilitant ainsi sa mise en production. Ce chapitre marque donc une avancée significative vers la réalisation pratique de nos objectifs.

## 1. Outils de développement

### 1.1.Outils Technologiques :

- **React js** : est une bibliothèque JavaScript open source conçue pour développer des interfaces utilisateur interactives et performantes, principalement pour les applications web single-page et les applications mobiles via React Native. Elle repose sur une architecture de composants réutilisables qui encapsulent leur propre état et leur logique de rendu, facilitant ainsi la maintenance et la scalabilité du code. React utilise un DOM virtuel qui optimise les performances en calculant les modifications nécessaires avant de mettre à jour le DOM réel, et propose JSX comme syntaxe permettant d'écrire des éléments HTML directement dans le code JavaScript. Ses principales fonctionnalités incluent la gestion d'état avec les hooks, le cycle de vie des composants, le rendu conditionnel, et un écosystème riche d'outils et de bibliothèques complémentaires qui en font l'une des solutions les plus populaires pour le développement front-end moderne.
- **Chart.js** : est une bibliothèque JavaScript open source spécialisée dans la création de graphiques interactifs et responsives pour les applications web. Elle propose une variété de types de graphiques incluant les courbes, barres, secteurs, radars, et graphiques en aires, tous personnalisables via des options de configuration flexibles. Basée sur le canvas HTML5, Chart.js génère des visualisations fluides et animées qui s'adaptent automatiquement aux différentes tailles d'écran. La bibliothèque est facile à intégrer, légère en termes de performances, et offre des fonctionnalités avancées comme les tooltips interactifs, les légendes personnalisables et les mises à jour dynamiques des données en temps réel.
- **React Bootstrap** : est une bibliothèque qui adapte Bootstrap aux applications React en fournissant des composants UI pré-construits comme les grilles, boutons, modales et formulaires. Elle élimine la dépendance à jQuery et offre une intégration native avec





React, permettant aux développeurs d'utiliser les éléments Bootstrap comme des composants React standard avec props et gestion d'état.

- **Bootstrap** : est un framework CSS open source qui fournit une collection de styles, composants et outils JavaScript prêts à l'emploi pour développer rapidement des interfaces web responsives et modernes. Il propose un système de grille flexible, des composants UI standardisés comme les boutons, formulaires, barres de navigation et modales, ainsi que des utilitaires CSS pour la mise en page et le style. Bootstrap permet aux développeurs de créer des sites web adaptatifs qui s'ajustent automatiquement aux différentes tailles d'écran, réduisant considérablement le temps de développement front-end.
- **PHP** : est un langage de programmation open source spécialement conçu pour le développement web côté serveur. Il permet de créer des pages web dynamiques en s'intégrant facilement dans le code HTML et en interagissant avec les bases de données comme MySQL. PHP s'exécute sur le serveur avant d'envoyer le résultat au navigateur, offrant des fonctionnalités pour la gestion des formulaires, sessions utilisateur, manipulation de fichiers et création d'APIs. Grâce à sa simplicité d'apprentissage et sa large compatibilité avec les serveurs web, PHP est devenu l'un des langages les plus utilisés pour le développement d'applications web et de systèmes de gestion de contenu.
- **MySQL** : est un système de gestion de base de données relationnelle (SGBDR) open source qui utilise le langage SQL pour la manipulation et l'interrogation des données. Il permet de stocker, organiser et récupérer efficacement de grandes quantités d'informations structurées dans des tables liées entre elles par des relations. MySQL offre des fonctionnalités avancées comme la gestion des transactions, la réplication, la sécurité des données et l'optimisation des performances. Largement utilisé dans le développement web en combinaison avec PHP, MySQL est reconnu pour sa fiabilité, sa rapidité et sa facilité d'administration, ce qui en fait l'une des bases de données les plus populaires pour les applications web et les systèmes d'information.

## 1.2.Outils de base

- **Visual Studio**

Visual Studio est un outil de développement puissant qui permet d'effectuer l'ensemble du cycle de développement au même endroit. Il s'agit d'un environnement de développement intégré (IDE) complet permettant d'écrire, de modifier, de déboguer et de générer du code, puis de déployer votre application. En plus de l'édition et du débogage du code, Visual Studio comprend des compilateurs, des outils de complétion de code, un contrôle de code source, des extensions et de nombreuses autres fonctionnalités qui améliorent chaque étape du processus de développement logiciel.

- **Git Hub :**

GitHub est une plateforme de développement collaboratif basée sur Git. Git est un système de contrôle de version, et GitHub utilise ce système pour permettre aux développeurs de travailler ensemble sur des projets logiciels. Sur GitHub, les utilisateurs peuvent partager, collaborer et contribuer à des projets, qu'il s'agisse de projets open source ou de projets privés.

## 1.4. Tests et Validation

- **Postman :** est un outil de développement et de test d'APIs qui permet aux développeurs de créer, tester, documenter et surveiller des interfaces de programmation applicatives de manière intuitive. Il offre une interface graphique conviviale pour envoyer des requêtes HTTP (GET, POST, PUT, DELETE) vers des serveurs, examiner les réponses, et valider le comportement des APIs. Postman propose des fonctionnalités avancées comme l'organisation des requêtes en collections, l'automatisation des tests avec des scripts, la gestion des environnements de développement, et la collaboration en équipe. Cet outil est devenu indispensable pour le développement d'applications web modernes, facilitant le processus de développement, de débogage et de documentation des services web.

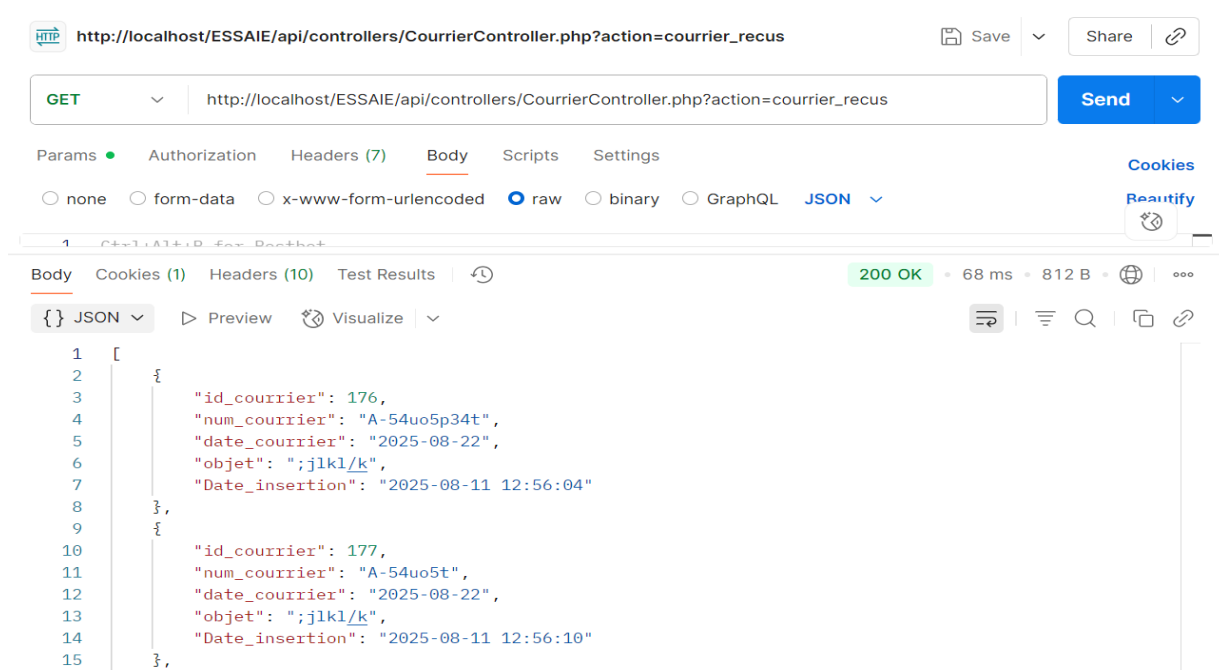


Figure 7 | Endpoint API de réception des courriers (courrier\_recus) avec réponse JSON et en-têtes d'autorisation

## 2. Présentation de l'application

- Page d'authentification.
- Espace Agent.
- Espace Responsable.
- Dashboard administrateur pour :

### • Authentification



Figure 8 : Informations invalides et message d'erreur

- Espace agent :

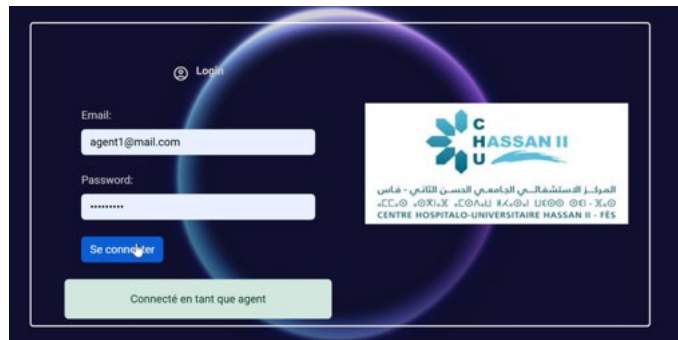


Figure 9 : Authentification tant qu'agent



Figure 10 : Interface d'utilisateur agent

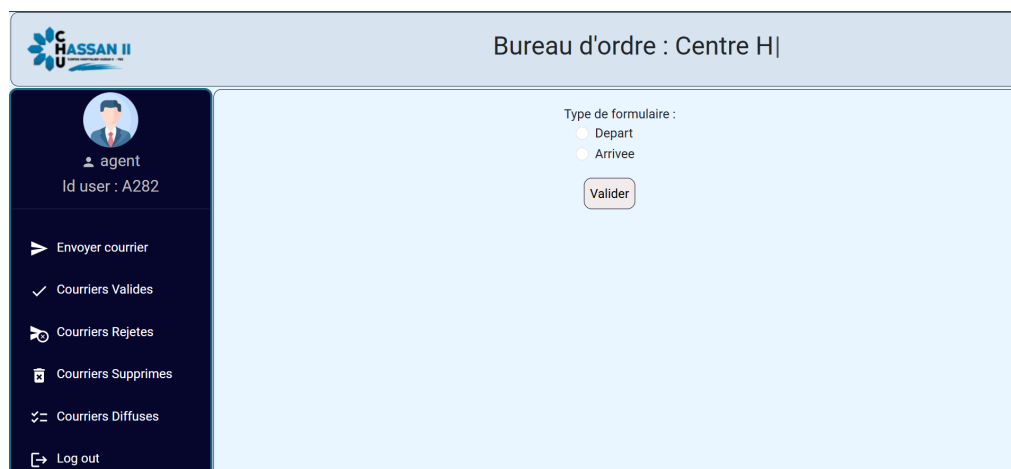


Figure 11 : Choix de type de courrier



**Formulaire de depart :**

Numero de courrier:

Date de courrier:

Objet:

Fichier PDF:    
Formats acceptés : PDF uniquement

Emetteur (Unite interne):

Destinataire: 

-- Unites internes --  
**Divisions:**  
Division des Affaires Financières  
Division d'Approvisionnement et  
Division des Ressources Humain

-- Unites externes --  
Ministère de la Santé  
Agence Nationale des Médicaments  
Organisme National d'Assurance

Maintenez Ctrl (Windows) ou ⌘ (Mac) pour sélectionner plusieurs options

Figure 12 : Forme de formule de départ

Numero de courrier:

Date de courrier:

Objet:

Fichier PDF:    
Formats acceptés : PDF uniquement

Emetteur (Unite interne):

Destinataire: 

Hôpital des Spécialités  
Hôpital Mère-Enfant  
Hôpital d'Oncologie  
Hôpital Omar Driss  
Hôpital Ibn Al Hassan

Ministère de la Santé  
Agence Nationale des Médicaments  
Organisme National d'Assurance  
Institut Pasteur  
Organisme National d'Assurance

Maintenez Ctrl (Windows) ou ⌘ (Mac) pour sélectionner plusieurs options

Champs manquants: Numéro Date Objet

Figure 13: Message d'erreurs si informations invalides



Numero de courrier:

Date de courrier:

Objet:

Fichier PDF:    
Formats acceptés : PDF uniquement

Emetteur (Unite interne):

Destinataire: 

Division des Systemes d'Inform  
Division Recherche et Innovation  
**Sections**  
Section Audit Interne  
Section Contrôle de Gestion

Ministère de la Santé  
Agence Nationale des Médicam  
Organisme National d'Assurance  
Institut Pasteur  
Organisme National

Maintenez Ctrl (Windows) ou ⌘ (Mac) pour sélectionner plusieurs options

**Succès**

Figure 14: Message de succès si courrier est diffuse

**Formulaire de courrier d'arrivee :**

Numero de courrier:

Date de courrier:

Objet:


Fichier PDF:    
Formats acceptés : PDF uniquement

Emetteur:


Chosir un seul emetteur (soit interne ou externe)

**Succès**

Figure 15: Forme de formulaire de courrier d'arrivée



Bureau d'ordre : Centre Hos|



agent  
Id user : A282

- Envoyer courrier
- ✓ Courriers Valides
- ✗ Courriers Rejetés
- 🗑 Courriers Supprimés
- 📢 Courriers Diffusés
- ➡ Log out

### Courriers Valides

ID	Numéro	Date	Objet	Date d'insertion	
194	A-563457437	2025-09-10	objet 9564570546	2025-08-14 18:40:22	<button>Ouvrir</button>
195	A-56345894093	2025-09-10	objet r83725p2849o	2025-08-14 18:40:44	<button>Ouvrir</button>
202	A-56494fdfed	2025-01-22	objet lorem r494004 efwdw	2025-08-14 18:49:49	<button>Ouvrir</button>

Figure 16: Affichage de courriers valides

**Courrier de numero : A-563457437**

Numero de courrier:

A-563457437

Date de courrier:

10/09/2025

Objet:

objet 9564570546

Ouvrir le PDF

Emetteur:

Hôpital des Spécialités

Recepteur internes

Division des Affaires Financières  
Division d'Approvisionnement et Logistique

Recepteur Externes

Ministère de la Santé  
Agence Nationale des Médicaments

Figure 17: Affichage de courrier valide

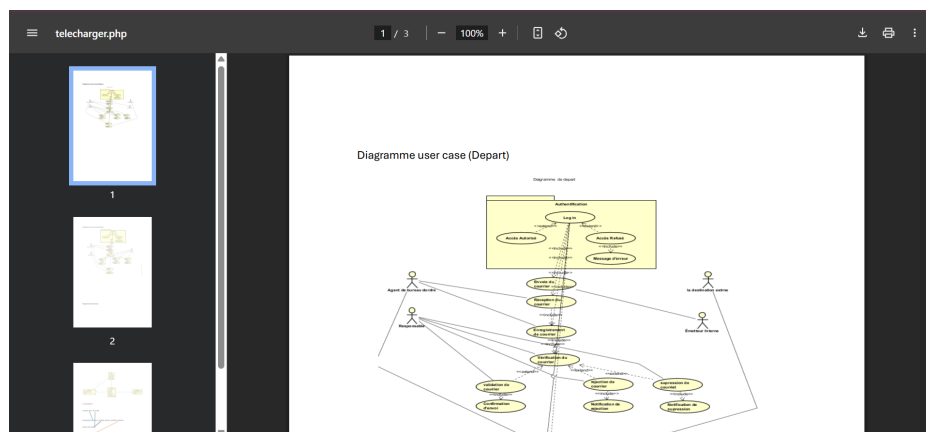


Figure 18: Affichage de pdf de courrier



Courriers Supprimés  
Courriers Diffusés  
Log out

Emetteur: Hôpital des Spécialités

Recepteur internes: Division des Affaires Financières  
Division d'Approvisionnement et Logistique

Recepteur Externes: Ministère de la Santé  
Agence Nationale des Médicaments

Annotations : Anootalon 9834024

Diffuser Renitialiser

Courrier diffuse

Figure 19: Diffusion d'un courrier valide

Bureau d'ordre : Centre Hospitalier Unil

agent  
Id user : A282

Envoyer courrier

Courriers Valides

Courriers Rejetés

Courriers Supprimés

Courriers Diffusés

Log out

### Courriers Rejets

ID	Numéro	Date	Objet	Date insertion	Date rejet	Motif
178	A-54uot	2025-08-22	ijkl/k	2025-08-11 12:56:14	2025-08-11 12:57:43	Afficher
179	A-54utt	2025-08-22	ijkl/k	2025-08-11 12:56:19	2025-08-11 13:00:56	Afficher
182	A-54uttj	2025-08-22	ijkl/k	2025-08-11 12:56:52	2025-08-11 23:59:13	Afficher
183	A-54uttjij	2025-08-22	ijkl/k	2025-08-11 12:57:12	2025-08-11 23:47:59	Masquer
190	D-i34 rejetes	2025-08-08	jhlj	2025-08-12 10:50:58	2025-08-12 10:54:09	Afficher
193	A-0g3kng	2025-08-24	objet 37853	2025-08-14 11:35:01	2025-08-14 11:40:39	Afficher

Figure 20: Affichage des courriers rejetés

Bureau d'ordre : Centre Hospitalier Universita|

agent  
Id user : A282

Envoyer courrier

Courriers Valides

Courriers Rejetés

Courriers Supprimés

Courriers Diffusés

Log out

### Courriers Supprimés

ID	Numéro	Date	Objet	Date de suppression
20	A-54uttj	2025-08-22	ijkl/k	2025-08-11 23:48:06
21	A-54uoSp34	2025-08-22	ijkl/k	2025-08-12 10:20:12
24	A-io34i34	2025-08-21	iojlnl	2025-08-12 10:39:43
25	D0iqryw supprimes	2027-06-12	knsfmsdl lgilfpjwfrga	2025-08-12 10:54:17
26	A-5649ddw4wdfdfed	2025-01-22	objet loredm r494004 efwdw	2025-08-15 09:22:26
27	A-5649d4wdfdfed	2025-01-22	objet loredm r494004 efwdw	2025-08-26 11:14:27
28	wrt476erhdr	2025-08-08	ijkl/k	2025-09-02 13:35:39

Figure 21: Affichage des courriers supprimés



Bureau d'ordre : Centre Hospi|

Courriers Diffuses

ID	Numéro	Date	Objet	Type de courrier	Date d'insertion	Annotation
159	D192o3	2025-08-14	iojlnl	depart	2025-08-11 12:39:53	
160	D192o3j	2025-08-14	iojlnl	depart	2025-08-11 12:40:34	
162	D192o3jj	2025-08-14	iojlnl	depart	2025-08-11 12:40:54	
164	D192o3jhh	2025-08-14	iojlnl	depart	2025-08-11 12:44:20	
165	D192o3jjhd	2025-08-14	iojlnl	depart	2025-08-11 12:45:42	

Figure 22: Affichage des courriers diffuses

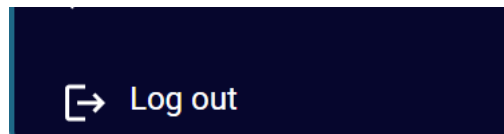


Figure 23: Button de Log out

- Espace responsable :

Login

Email: respo2@mail.com

Password: .....

Se connecter

Connecté en tant que responsable

Figure 24: Connexion tant que responsable

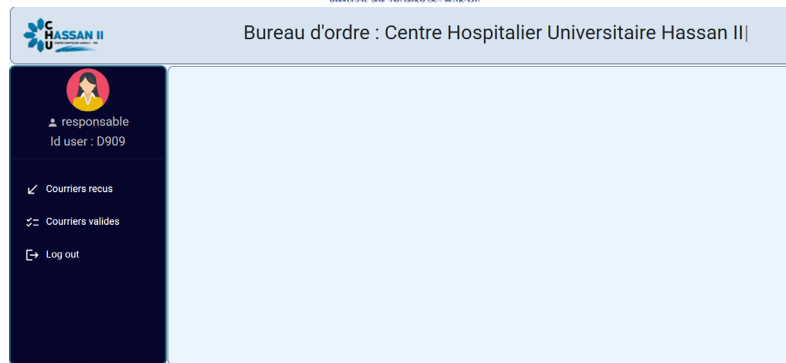


Figure 25: Interface responsable

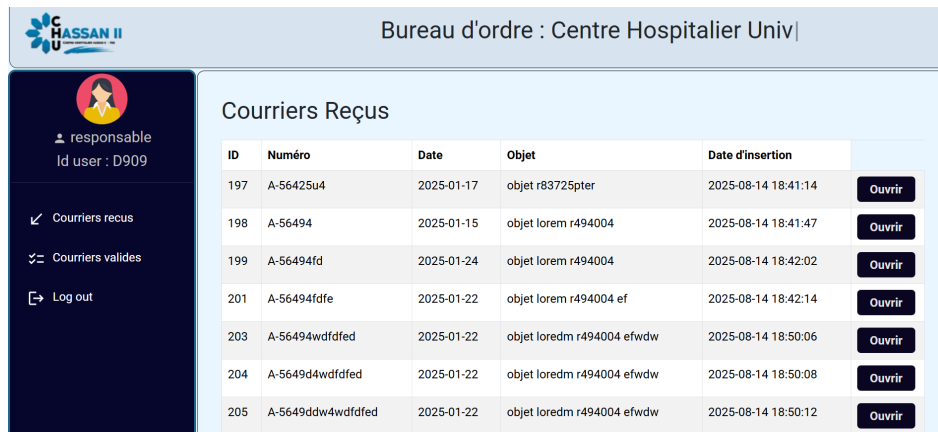


Figure 26: Affichage des courriers reçus



Figure 27: Affichage d'un courrier reçu



Objet: objet lorem r494004 efwdw

[Ouvrir le PDF](#)

Emetteur: OMS Bureau Régional

Destinataire: 

Sections  
Section Audit Interne  
Section Contrôle de Gestion  
Section Management de la Qua  
Section Communication et Co

Unités externes –  
Ministère de la Santé  
Agence Nationale des Médica  
Organisation National d'Assura

Maintenez Ctrl (Windows) ou ⌘ (Mac) pour sélectionner plusieurs options

Annotations : Annotation djsls

[Valider](#) [Rejeter](#) [Supprimer](#) [Renitialiser](#)

Validation complète réussie !

Figure 28: Validation du courrier reçu

[Ouvrir le PDF](#)

Emetteur: OMS Bureau Régional

Destinataire: 

– Unites internes –  
Divisions:  
Division des Affaires Financière  
Division d'Approvisionnement e  
Division des Ressources Humaine

– Unites externes –  
Ministère de la Santé  
Agence Nationale des Médica  
Organisation National d'Assura

Maintenez Ctrl (Windows) ou ⌘ (Mac) pour sélectionner plusieurs options

Annotations : ....

Motif rejection : Motif de rejection .....

[Valider Rejection](#) [Annuler](#)

[Valider](#) [Rejeter](#) [Supprimer](#) [Renitialiser](#)

Figure 29: Rejection du courrier reçu



Ouvrir le PDF

Emetteur: OMS Bureau Régional

Destinataire:   
 - Unites internes -   
 Divisions:   
 Division des Affaires Financière   
 Division d'Approvisionnement e   
 Division des Ressources Humaines   
 - Unites externes -   
 Ministère de la Santé   
 Agence Nationale des Médicaments   
 Agence Nationale d'Assurance

Maintenez Ctrl (Windows) ou ⌘ (Mac) pour sélectionner plusieurs options

Annotations : ....

Valider Rejeter Supprimer Renitialiser

Supression Valide !

Figure 30: Suppression du courrier reçu

Bureau d'ordre : Centre Hospitalier Universita

Courriers Valides

ID	Numéro	Date	Objet	Date d'insertion
203	A-56494wdfded	2025-01-22	objet loredm r494004 efwdw	2025-08-14 18:50:06

responsable   
 Id user : D909

← Courriers recus   
 ↗ Courriers valides   
 ➡ Log out

Figure 31: Affichage des courriers valides

## • Espace Admin :

Logo

Email:   
 admin1@mail.com

Password:   
 \*\*\*\*\*

Se connecter

Connecté en tant que admin

C HASSAN II   
 U   
 المركز الاستشفائي الجامعي الحسن الثاني - فاس   
 ⵜⴰⵎⴰⵔⵜ ⵜⴰⵎⴰⵏⴰⵢⵜ ⵜⴰⵏⴰⵢⵜ ⵜⴰⵏⴰⵢⵜ ⵜⴰⵏⴰⵢⵜ   
 CENTRE HOSPITALO-UNIVERSITAIRE HASSAN II - FES

Figure 32: Connexion tant qu'Admin

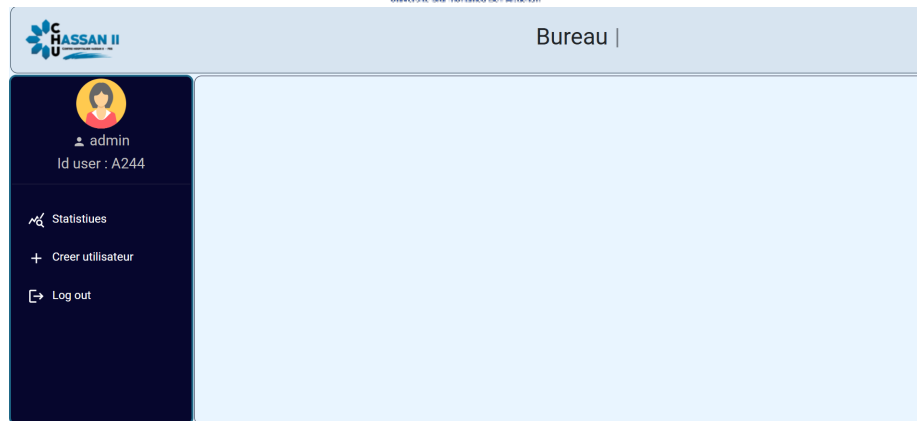


Figure 33: Interface Admin

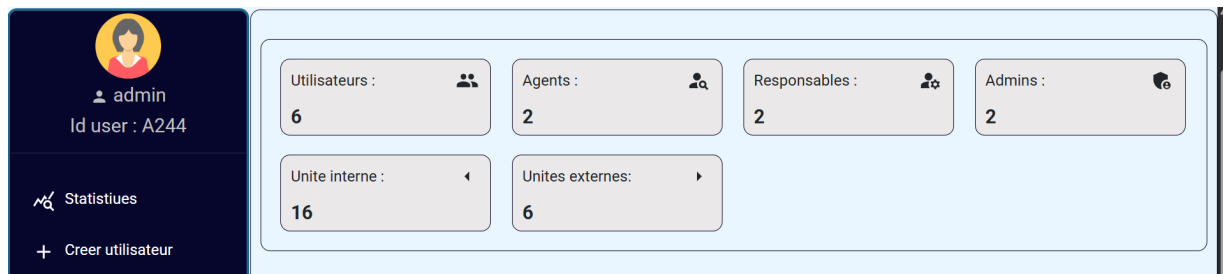


Figure 34: Statistiques 1

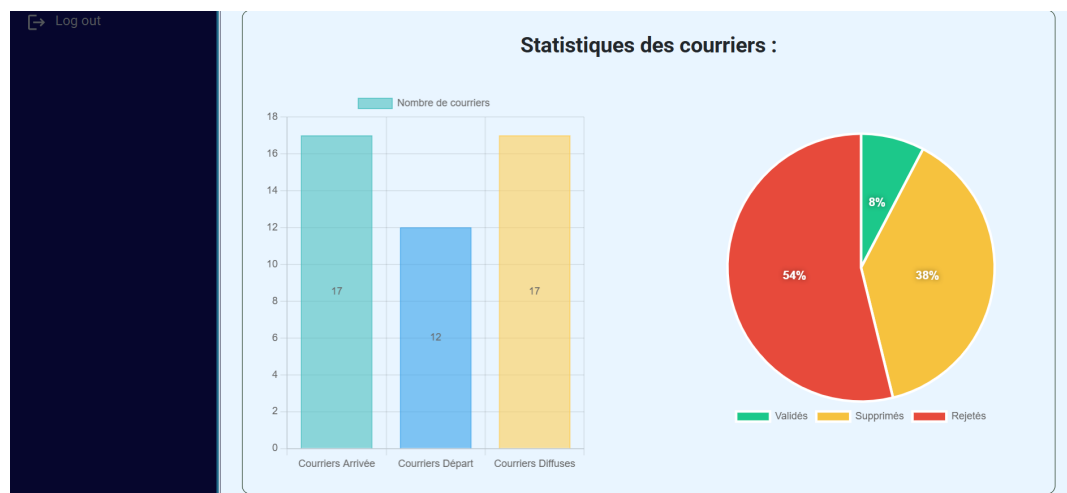


Figure 35: Statistiques 2



The screenshot shows a web application interface for creating a user. On the left is a dark sidebar with a user profile (admin, Id user: A244) and navigation links: Statistiques, Creer utilisateur, and Log out. The main area is titled 'Formulaire de creation d'un utilisateur :'. It contains several input fields: 'Non:' (Jeann), 'Prenon:' (Aloca), 'Email:' (reglkjweijg@gamil.com), 'Password:' (jnweoriuq29), 'Unite interne:' (Hôpital Omar Driss), 'Service:' (Contrôle de Gestion), and 'Role:' (Agent). At the bottom, a green message box states 'Utilisateur cree par succes'.

Figure 36: Création de user par succès

The screenshot shows the same user creation form as Figure 36, but with invalid data. The 'Non:' field contains 'yassime', 'Prenon:' contains 'efefef', and 'Email:' contains a long, nonsensical string. The 'Password:' field is empty. The 'Unite interne:' field is 'Section Pharmacie Centrale', 'Service:' is a dropdown menu showing '-- Choisissez --', and 'Role:' is 'Agent'. At the bottom, a red message box states 'Champs manquants: Mot de passe Service'.

Figure 37: Information invalides et message d'erreur



### 3. Le modèle MVC (Modèle-Vue-Contrôleur)

#### 1. Modèle (PHP)

- Gère les données et la logique métier.
- Interagit avec la base de données (ex: requêtes SQL via PDO/MySQLi).
- Expose des API REST (routes PHP) pour que React puisse fetch les données (ex: avec fetch() ).

#### 2. Vue (React)

- Affiche l'interface utilisateur (JSX).
- Reçoit les données du backend via des appels API (GET/POST).
- Pas de logique métier : juste l'affichage et les interactions UI.

#### 3. Contrôleur (PHP )

- Le PHP agit comme **contrôleur** : il reçoit les requêtes HTTP, traite la logique métier et renvoie des réponses (généralement en JSON pour React).
- Exemple : Un fichier api.php qui gère les routes (/get-users, /add-post).

### API RESTful : Définition

Une API RESTful est une interface qui permet à deux systèmes (ex: un frontend React et un backend PHP) de communiquer via des requêtes HTTP/HTTPS, en suivant des principes clés :

- Méthodes standard : GET (lire), POST (créer), PUT (mettre à jour), DELETE (supprimer).
- Stateless : Chaque requête est indépendante (le serveur ne garde pas d'état).
- URLs structurées : Ex: /api/ressources/{id}.
- Format de données : Généralement JSON (léger et lisible).

### Conclusion

En définitive, ce chapitre a établi le pont essentiel entre les fondements théoriques et l'implémentation concrète de ma solution applicative. Les décisions technologiques que j'ai prises se sont révélées cruciales pour assurer une interface utilisateur performante et un processus de développement fluide. L'incorporation des mécanismes de tests unitaires et des procédures de déploiement a renforcé la fiabilité et la durabilité de l'application. Ces réalisations me conduisent avec confiance vers l'aboutissement de mon projet.



## Conclusion Générale

En conclusion, le développement de cette Application Web de Gestion des Courriers de Bureau d'Ordre a constitué une réponse pertinente aux enjeux de centralisation des flux documentaires du CHU Hassan II de Fès. Basée sur React pour l'interface utilisateur, Bootstrap pour la mise en forme, PHP pour le backend et MySQL pour la base de données, cette solution a prouvé son efficacité dans l'amélioration de la coordination inter-services hospitaliers.

L'architecture que j'ai développée, structurée par une modélisation UML rigoureuse avec les diagrammes de classes et de cas d'utilisation, offre une base robuste capable de s'adapter aux besoins organisationnels. Cette approche méthodologique m'a permis d'optimiser le traitement des courriers entrants et sortants, de consolider la traçabilité des documents et d'instaurer une transparence accrue dans les processus administratifs.

Les fonctionnalités implémentées - gestion des profils, enregistrement et suivi des courriers - ont généré des bénéfices observables : diminution notable des délais de traitement documentaire et amélioration substantielle du suivi, éléments fondamentaux pour le fonctionnement optimal de l'établissement de santé. L'application centralisée facilite désormais la coordination entre les diverses divisions du CHU, réduisant les risques d'erreurs et les retards inhérents au système manuel précédent.

Ce projet de stage d'initiation représente une contribution significative à la modernisation de la gestion administrative hospitalière, proposant une solution pérenne et adaptable. Il illustre l'importance d'une démarche structurée, conjuguant modélisation conceptuelle et développement pratique, dans la création d'applications web performantes.

Cette expérience a constitué un enrichissement personnel considérable, me permettant de concrétiser mes acquis académiques dans un environnement professionnel réel, tout en développant des compétences pratiques en gestion de projet et en développement d'applications web, apportant ainsi une valeur ajoutée tangible à l'administration du Centre Hospitalier Universitaire Hassan II.





## Bibliographie et Webographie

- Astah UML — <https://astah.net> (consulté le **05 juillet 2025**)
- React — <https://react.dev> (consulté le **07 juillet 2025**)
- React Bootstrap — <https://react-bootstrap.github.io> (consulté le **08 juillet 2025**)
- Bootstrap — <https://getbootstrap.com> (consulté le **09 juillet 2025**)
- Chart.js — <https://www.chartjs.org> (consulté le **10 juillet 2025**)
- PHP — <https://www.php.net/docs.php> (consulté le **12 juillet 2025**)
- MySQL — <https://dev.mysql.com/doc> (consulté le **14 juillet 2025**)
- Postman — <https://www.postman.com> (consulté le **16 juillet 2025**)