

# *Dédicaces*

*A ma chère et tendre mère Leila,  
La femme qui m'a tout offert pour réussir.*

*A mon cher père Mohamed,  
L'homme à qui je dois tout le respect.  
Que Dieu m'aide à vous rendre un petit peu de vos  
sacrifices.*

*A mes sœurs Imen et Chiraz, mon frère Chafik et ma  
copine Khawla,  
En témoignage de mon profond amour et respect, à qui  
je souhaite  
le succès et le bonheur.*

*A mes chers amis, à tous ceux que j'aime, tous ceux qui  
m'aiment  
Et tous ceux qui me sont chers...*

*Ghazi Toujani...✍*

# REMERCIEMENTS

*Je tiens au début de ce rapport à exprimer mes vifs remerciements ainsi que toute ma gratitude à tous ceux qui m'ont aidé durant ce stage ingénieur que j'ai effectué au siège technique de **BIAT** et qui m'a été autant instructif que bénéfique.*

*Je remercie spécialement pour ses supports logistiques **Monsieur Moncef MALLEK** et **Monsieur Amir KTARI**, qui m'ont offert l'opportunité d'effectuer ce stage, pour ses vifs esprits de collaboration ainsi que ses précieux conseils et recommandations.*

*Je voudrais également exprimer ma vive reconnaissance et présenter mes sincères remerciements à l'ensemble du personnel du centre de **BIAT** pour leur accueil ainsi que l'encouragement dont ils m'on fait part, mais aussi pour leur aide précieuse à réaliser ce travail.*

## Table des matières

Introduction générale.....	6
Chapitre 1 – Cadre et Contexte du projet .....	7
Introduction .....	7
1. Présentation générale.....	7
1.1. Présentation de la banque [1].....	7
1.2. Organigramme de la banque <sup>[2]</sup> .....	8
2. Présentation du département d'accueil <sup>[2]</sup> .....	8
2.1. Le domaine « Migration » <sup>[2]</sup> .....	9
2.2. Organigramme du département d'accueil : Coordination Études & Développements SI <sup>[2]</sup> .....	10
3. Cadre et contexte du stage .....	11
4. Choix de la méthode .....	11
5. Choix du langage de modélisation .....	13
Conclusion .....	13
Chapitre 2 – Spécification des besoins et la conception.....	14
Introduction .....	14
1. Besoins fonctionnels .....	14
2. Besoins non fonctionnels.....	15
3. Analyse des besoins .....	16
3.1. Diagramme de cas d'utilisation général .....	16
3.2. Diagramme de séquence « Authentification » .....	19
3.3. Diagramme de séquence « générer des rapports ».....	20
3.4. Diagramme de classes .....	21
Conclusion .....	21
Chapitre 3 – Réalisation.....	22
Introduction .....	22
1. Environnement de travail.....	22
1.1. Environnement matériel.....	22
1.2. Environnement logiciel.....	22
1.3. Plate-forme applicative .....	23
1.4. Framework <sup>[12]</sup> .....	24
2. Aperçus & Description .....	25
Conclusion .....	26

Conclusion Générale.....	27
Bibliographie et Webographie .....	28

# Liste des figures

Figure 1.1 : Organigramme de la BIAT .....	8
Figure 1.2 : Organigramme du département « Coordination Etudes & Développements SI » .....	10
Figure 1.3 : Fonctionnement SCRUM.....	12
Figure 1.4 : Diagramme de cas d'utilisation général.....	16
Figure 1.5 : Diagramme de séquence authentification .....	19
Figure 1.6 : Diagramme de séquence générer des rapports .....	20
Figure 1.7 : Diagramme de classes.....	21
Figure 1.8 : Interface d'authentification.....	25
Figure 1.9 : Interface d'inscription d'un utilisateur.....	25
Figure 1.10 : Interface de gestion du base de donnée.....	26

# Introduction générale

Maintenant que les systèmes d'information sont désormais primordiaux pour toute activité de la société moderne, il est bien établi que vendre et réaliser un chiffre d'affaire important s'avère insuffisant pour la survie d'une entreprise.

La gestion moderne des entreprises impose désormais une maîtrise de l'information et une stratégie de développement à moyen et à long terme.

Une base de données est une entité dans laquelle il est possible de stocker des données de façon structurée et avec le moins de redondance possible. Ces données doivent pouvoir être utilisées par des programmes, par des utilisateurs différents. Ainsi, la notion de base de données permet de mettre des données à la disposition d'utilisateurs pour une consultation ou bien une manipulation, tout en s'assurant des droits accordés à ces derniers.

C'est dans ce cadre que s'inscrit notre stage d'été où les utilisateurs du BIAT utilisent les données des bases de données pour créer des rapports dans le but des besoins spécifiques pour chaque utilisateur.

Enfin, nous avons dû réaliser une application Web qui permettra de faire le reporting comptable à fin de générer des rapports à partir des données enregistrés dans la base de données.

Le présent rapport sera organisé en trois chapitres.

Le premier chapitre présentera l'entreprise BIAT et le cadre du stage.

Le deuxième chapitre entamera dans une première partie la spécification des besoins, et dans une seconde partie la conception.

Alors que le troisième chapitre portera sur les principales interfaces de notre application Web.

# **Chapitre 1 – Cadre et Contexte du projet**

## **Introduction**

Dans ce chapitre, nous allons commencer par présenter dans une première section l'entreprise où s'est déroulé notre stage d'été et décrire ses principales activités. Dans une seconde étape, nous allons présenter le cadre proprement dit du stage.

## **1. Présentation générale**

### **1.1. Présentation de la banque [1]**

La BIAT est une banque qui existe depuis Avril 1976. C'est une société anonyme qui a choisi la gestion à directoire et conseil de surveillance pour revenir à partir de Mai 2005 à une gestion classique. Son capital est de 100.000.000 TND divisé en 10000000 actions, de nominal 10 dinars l'une.

Créée en 1976, la Banque Internationale Arabe de Tunisie « BIAT » est le résultat de l'association entre des initiatives, des capitaux tunisiens principalement du secteur privé et des efforts d'institutions financières arabes et internationales.

Pour atteindre les objectifs économiques et sociaux qu'elle s'est assignés dès le départ, la BIAT a tenu à mettre en place un système d'information performant lui permettant une gestion transparente et indépendante pour un développement soutenu et rapide.

En effet, la BIAT est une banque relativement jeune si on avait à la comparer à certaines autres institutions du système bancaire.

La gestion des ressources humaines a été une préoccupation constante des dirigeants de la BIAT. C'est pourquoi elle se dote dès les premières années de sa création de sa propre structure de formation. Elle fait appel tant aux hauts cadres de la banque qu'à des compétences de l'extérieur, pour dispenser les différents modules de formation fixés en adéquation avec l'évolution du métier bancaire.

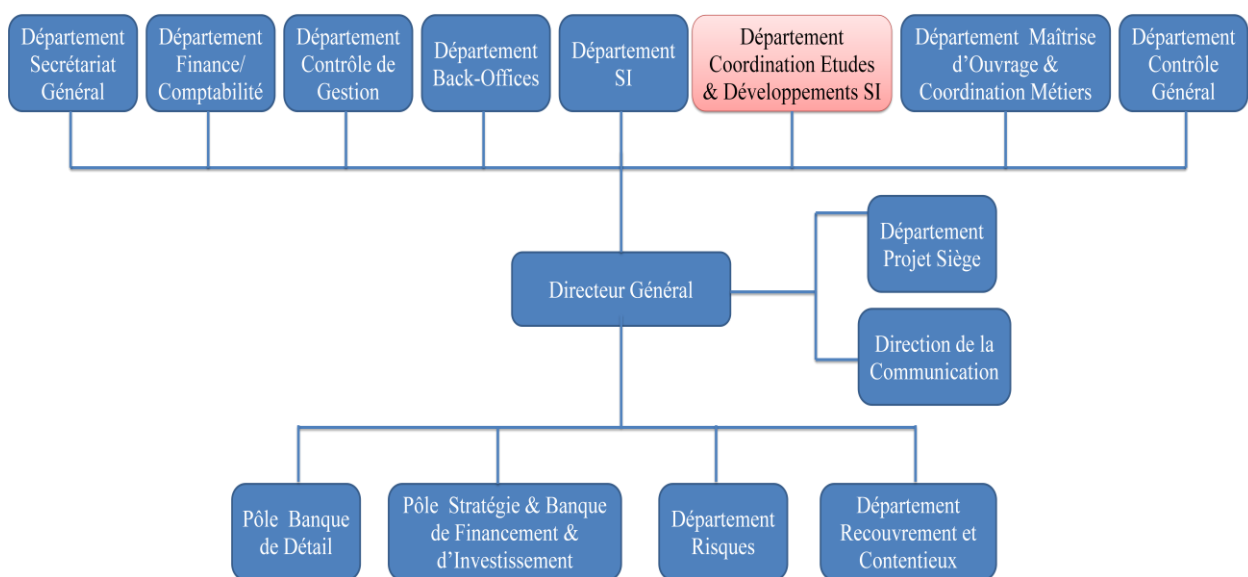
Cependant, la BIAT demeure, malgré sa position, en plein chantier de refonte de son système d'information. Cette refonte vise à innover son système afin d'être à la hauteur de

leur institution. Pour cela, la BAT a fait appel à plusieurs sociétés étrangères ainsi qu'à son personnel qualifié pour innover son système d'information.

## 1.2. Organigramme de la banque <sup>[2]</sup>

La BIAT, comme tout organisme de grande taille, est dotée d'un organigramme qui décrit et schématise son squelette. Ce dernier nous permet de comprendre les interactions existantes entre les différents départements qui le composent.

Nous présentons dans la **figure 1.1** l'organigramme de la BIAT :



**Figure 1.1** : Organigramme de la BIAT

Le directeur général a sous sa tutelle l'ensemble des départements qui composent la banque. Parmi ces départements nous pouvons citer celui de la « Coordination Etudes et Développement SI » au sein duquel nous avons effectué notre stage.

## 2. Présentation du département d'accueil <sup>[2]</sup>

Le département « Coordination Études et Développement SI » est chargé de :

- ✓ Définir l'urbanisation des référentiels et gérer les dictionnaires des données ;
- ✓ Assurer l'intégrité et la cohérence des données dans le système de production ;



- ✓ Développer les outils d'extraction des données du SI, construire et maintenir les entrepôts de données ;
- ✓ Mettre à disposition des métiers des solutions de restitution des données ;
- ✓ Assurer les migrations des données.

## **2.1. Le domaine « Migration » <sup>[2]</sup>**

Le domaine « Migration » assume la responsabilité de reprendre les données de l'ancien système (CTOS) pour les migrer dans le nouveau système (T24) et assure la planification et le bon déroulement de la bascule, le passage de CTOS à T24.

Les anciens systèmes d'exploitation installés sur les machines sont de type CTOS qui est un système d'exploitation qui date des années 80.

T24 est un système global bancaire, souple, évolutif et fonctionnant en temps réel. Il contribue à l'amélioration de la qualité des données par l'instauration de la notion du client, la centralisation des données, les contrôles instaurés... Il permet aussi d'assurer la cohérence de l'information et l'intégrité des données, d'avoir un référentiel unique et un accès facile à l'information.

L'enjeu de la migration est fondamental car les données sont le patrimoine de la banque. La migration permet de surmonter toutes les difficultés dues essentiellement à la qualité de certaines données, à leur disponibilité dans le système actuel et aux différences entre les deux systèmes CTOS et T24.

Ces deux systèmes sont totalement différents dans leur approche. Nous avons d'un côté le T24, un système global bancaire fonctionnant en temps réel avec un moteur comptable intégré et de l'autre le CTOS, entités déconnectées travaillant en différé avec le site central.

La direction « Référentiel SI et Décisionnel », au sein de laquelle nous avons effectué notre stage d'été, doit donc permettre la transformation et l'adaptation de l'ensemble des informations que vont s'échanger les deux systèmes.

Les différentes phases de la migration sont :

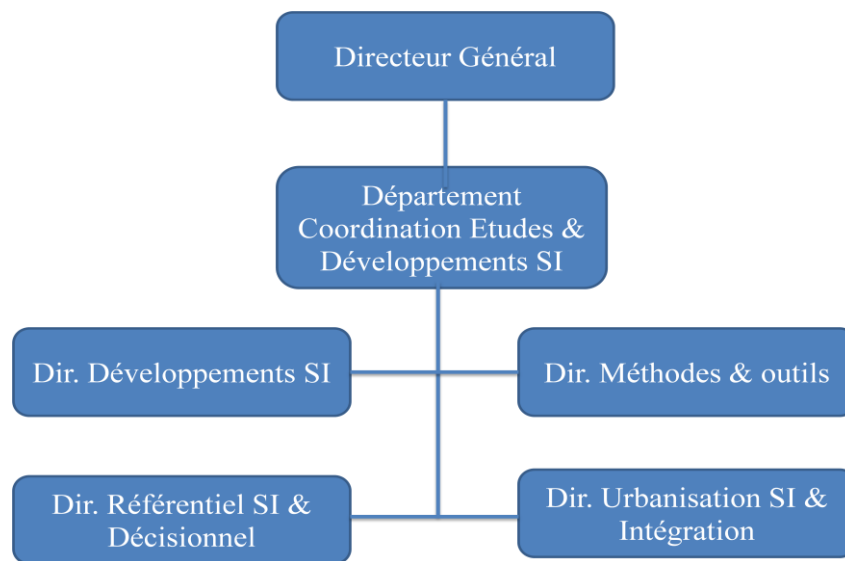
- ✓ L'analyse qui consiste à comprendre le fonctionnement de T24 et l'organisation de ces données, recenser les données dans le système CTOS à transférer sur T24, définir les

règles de reprise et à analyser la qualité des données. Cette dernière action peut engendrer des chantiers de fiabilisation.

- ✓ La construction qui consiste au développement des programmes d'implémentation des règles de reprise et des requêtes pour les indicateurs de contrôle.
- ✓ La certification statique qui permet de vérifier que les données sont bien migrées quantitativement et qualitativement.
- ✓ La certification dynamique qui consiste à dérouler des scénarios de test pour simuler le fonctionnement de T24 avec les données migrées.
- ✓ La définition et la mise en œuvre d'un plan de bascule qui précise le détail des opérations, la durée et les différents intervenants pour effectuer cette bascule du CTOS vers T24.

## 2.2. Organigramme du département d'accueil : Coordination Études & Développements SI <sup>[2]</sup>

Nous présentons dans la **figure 1.2** l'organigramme du département d'accueil :



**Figure 1.2 :** Organigramme du département « Coordination Etudes & Développements SI »

Cet organigramme présente les différentes directions rattachées au département « Coordination Etudes & Développements SI » qui est relié directement à la « Direction Générale ». Nous trouvons ainsi les directions « Développements SI », « Méthodes et Outils », « Urbanisation SI et Intégration » et « Référentiel SI et Décisionnel » au sein de laquelle nous avons effectué notre stage d'été.

### **3. Cadre et contexte du stage**

Notre stage s'inscrit dans le cadre d'un stage obligatoire pour les étudiants ayant accomplis leur deuxième année du cycle d'ingénieur à l'Institut Supérieur d'Informatique. Sa durée minimale est fixée par l'administration à un mois, et il doit être effectué dans une entreprise dont le domaine d'activité soit proche de la spécialité de l'étudiant ingénieur afin de mieux connaître le monde industriel, ce qui est très bénéfique pour faciliter l'intégration future dans le milieu du travail.

Pour notre stage, il s'est déroulé au sein de la Banque Internationale Arabe de Tunisie. Notre objectif est de mettre en place une application de reporting comptable : génération des rapports.

### **4. Choix de la méthode**

La méthode agile (en anglais Agile Modeling) est une approche itérative et incrémentale, menée dans un esprit collaboratif.

Ce type de méthode vise à réduire le cycle de vie du logiciel ; Ainsi nous débutons par développer une version minimale. Par la suite, nous intégrons les fonctionnalités adéquates selon un processus itératif basé sur une écoute client. L'ensemble du cycle de développement sera accompagné par des tests spécifiques. <sup>[3]</sup>

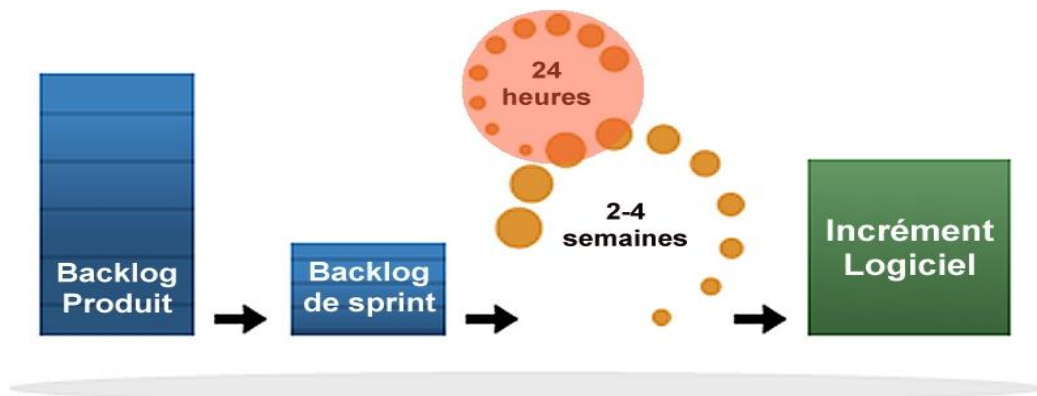
Les méthodes agiles ne sont pas une particularité de l'informatique décisionnelle, mais plutôt une nouvelle tendance dans la gestion des projets informatiques, qui répond aux difficultés inhérentes des méthodes traditionnelles et tendent à rapprocher le développeur du demandeur pour répondre plus rapidement et plus correctement à son besoin. <sup>[4]</sup>

Dans le cadre de son grand projet Horizon 2015, la BIAT souhaite motiver tous les collaborateurs (employés) en travaillant par objectif. Chaque employé se fixe des objectifs à atteindre avec son supérieur hiérarchique. Chaque année, un entretien annuel d'évaluation est fait pour noter chaque employé. Sa note globale est fixée selon l'atteinte ou non des objectifs qu'il s'est fixé l'année précédente. La méthode agile adoptée au sein de la BIAT est la

méthode scrum. C'est une méthode itérative et incrémentale qui permet de satisfaire au mieux les besoins du client et de maximiser les chances de réussite du projet.

Cette méthode met l'accent sur l'auto-organisation et l'auto-évaluation de l'équipe, le pouvoir de décision donné à l'équipe, les réunions quotidiennes et le planning adaptatif. Elle permet aussi de faire une démonstration du résultat de chaque itération appelée « sprint » afin de livrer un logiciel fonctionnel.

La méthode SCRUM est illustrée à travers le schéma de la **figure 1.3**.



**Figure 1.3 : Fonctionnement SCRUM**

Le Backlog produit (ou catalogue des besoins) est l'ensemble des fonctionnalités attendues d'un produit et qui sont évaluées par l'équipe. Il est élaboré avant le lancement des sprints, dans la phase de préparation plus connu sous « Sprint 0 ». Il est utilisé pour planifier la « Release » qui est une séquence de sprints ; Puis, à chaque sprint, lors de la réunion de planification de ce dernier pour décider du sous-ensemble qui sera réalisé. C'est donc un outil essentiel pour la planification. <sup>[5]</sup>

Le Backlog de sprint est extrait du backlog produit. Il correspond à l'ensemble des sprints où les besoins seront éclatés en tâches. Ces derniers consistent au développement des fonctionnalités du backlog de sprint. <sup>[5]</sup>

## **5. Choix du langage de modélisation**

La conception de notre projet est basée sur le concept objet et articulée autour d'UML (Unified Modeling Language) qui est par définition un langage de modélisation unifié et standardisé pour la conception des systèmes informatiques.

UML comble une lacune importante des technologies objet. Il permet d'exprimer et d'élaborer des modèles objet, indépendamment de tout langage de programmation. Il a été pensé pour servir de support à une analyse basée sur les concepts objet.

UML a été choisi car il n'impose pas de méthode de travail particulière. Il peut être intégré à n'importe quel processus de développement logiciel puisqu'il présente une sorte de boîte à outils qui permet d'améliorer progressivement les méthodes de travail tout en préservant les modes de fonctionnement.

Ce langage de modélisation est très intuitif, plus simple et cohérent que les autres méthodes. Il présente les différents points de vue de modélisation qui permettent de visualiser et manipuler les éléments de la modélisation. En plus, il permet de limiter de nombreux risques d'erreur.

## **Conclusion**

Dans ce chapitre, nous avons présenté l'entreprise, ses principales activités ainsi que le cadre du stage. Dans le chapitre suivant, nous allons entamer à l'analyse des besoins et la conception.

# Chapitre 2 – Spécification des besoins et la conception

## Introduction

La spécification des besoins est une première phase primordiale pour tout projet. Il s'agit, en effet, d'analyser les besoins du projet pour pouvoir identifier les fonctionnalités que le système doit assurer. Dans ce chapitre, nous allons détailler ce que cette application doit offrir à ses utilisateurs en termes de besoins fonctionnels et non fonctionnels ainsi que la conception qui constitue un pont entre la spécification et la réalisation.

## 1. Besoins fonctionnels

L'analyse fonctionnelle est une démarche qui consiste à caractériser les fonctions offertes par un produit pour satisfaire les besoins d'un utilisateur. En effet, chacun de ces besoins reflète les attentes des différents utilisateurs envers le système conçu. <sup>[6]</sup>

Nous exposons ainsi tous les besoins fonctionnels qui devront être implémentés au sein de notre projet à savoir :

- **Authentification** : Les utilisateurs doivent se connecter à travers un login et un mot de passe afin de bénéficier des fonctionnalités de l'application.
- **S'inscrire** : L'utilisateur peut faire une inscription pour pouvoir utiliser l'application.
- **Gérer les connections aux bases de données**: Après l'authentification de l'utilisateur, ce dernier a la possibilité d'ajouter, de modifier et de supprimer les connections.
- **Générer des rapports**: L'utilisateur peut génère un rapport à partir des données enregistrés dans les bases de données.
- **Se connecter à une base de données** : L'utilisateur à la possibilité de se connecter a n'importe qu'elle base de données choisie.

## 2. Besoins non fonctionnels

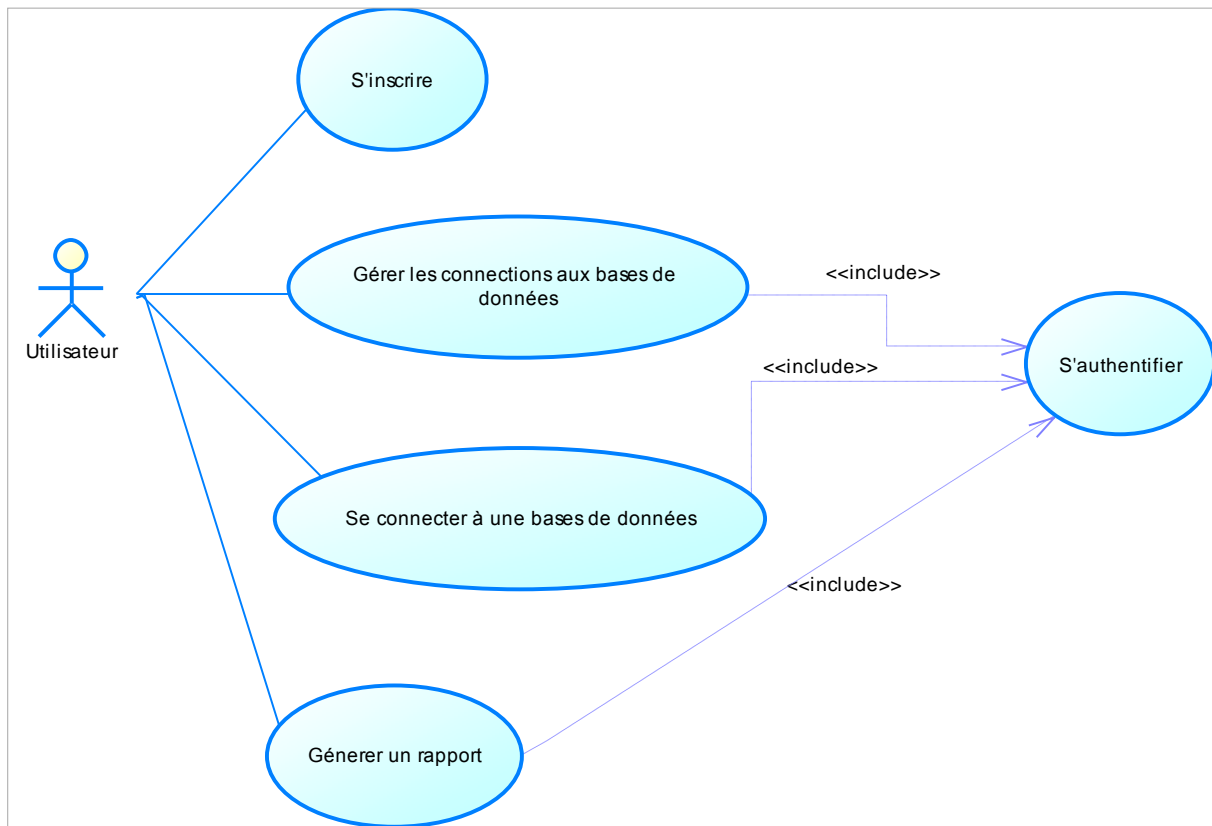
Les besoins non fonctionnels sont des contraintes que le produit doit impérativement vérifier, ou des fonctions complémentaires qui doivent être fournies pour l'utilisateur.

Il s'agit des besoins qui caractérisent le système à savoir :

- Rapidité : Il est clairement établi que la vitesse d'affichage influence fortement l'efficacité de l'application. C'est pour cette raison que nous avons développé une application fournissant des résultats efficaces et cohérents et dans un temps négligeable suite aux requêtes issues des utilisateurs.
- Ergonomie : l'application devra contenir des interfaces conviviales afin de faciliter son utilisation;
- Fiabilité : Les résultats apportés par l'application sont fiables et reflète effectivement l'état de la base au moment de son interrogation, c'est-à-dire des données mises à jour.
- Précision : Les résultats apportés par l'application doivent avoir le maximum de précision possible pour chaque type de résultat.
- Disponibilité : L'application doit être disponible à tout instant pour être utilisée par n'importe quel utilisateur, et doit être facilement accessible via n'importe quel ordinateur.
- Extensibilité : Toute amélioration ou modification doit être facile à réaliser, ainsi notre application devra garantir un degré d'extensibilité à tout développeur souhaitant y contribuer.

### 3. Analyse des besoins

#### 3.1. Diagramme de cas d'utilisation général



**Figure 1.4 :** Diagramme de cas d'utilisation général

#### Cas d'utilisation : s'authentifier

**Acteur primaire :** utilisateur

**Séquence principale :**

- 1- l'utilisateur demande à se connecter
- 2- le système affiche la page de login
- 3- l'utilisateur saisit son login et mot de passe et valide
- 4- le système valide les informations de login et mot de passe par rapport à celles qu'il a mémorisées
- 5- le système ramène l'utilisateur à la page de connexions aux bases de données.



**Séquence alternatives :**

- 1- lorsque l'utilisateur choisit de s'inscrire, le système déroule le cas d'utilisation « s'inscrire »
- 2- lorsque l'utilisateur fournit un login inconnu, le système affiche un message d'erreur et propose à l'utilisateur soit de saisir un autre login soit de s'inscrire.
- 3- lorsque le mot de passe est erroné, le système affiche un message d'erreur et demande à l'utilisateur de saisir à nouveau son mot de passe

**Cas d'utilisation : Gérer les connections aux bases de données**

**Acteur primaire :** utilisateur

**Pré-condition :** s'authentifier

**Séquence principale :**

- 1- lorsque l'utilisateur n'est pas connecté le système invoque le cas d'utilisation « s'authentifier »
- 2- l'utilisateur introduit les informations nécessaires (port, login, mot de passe, serveur et nom de serveur)
- 3- le système valide ces informations.

**Séquence alternative :**

- 1- lorsque l'utilisateur saisit les informations de la connexion, le système déroule le cas d'utilisation « ajouter connexion »
- 2- lorsque l'un de ces informations est incorrect, le système affiche un message d'erreur
- 3- quand l'utilisateur choisit une connexion dans la liste des connections à la possibilité de modifier les informations, le système déroule le cas d'utilisation « modifier connexion »
- 4- lorsque l'utilisateur choisit une connexion dans la liste des connections à la possibilité de la supprimer, le système déroule le cas d'utilisation « supprimer connexion »

### **Cas d'utilisation : Se connecter à une base de données**

**Acteur primaire :** utilisateur

**Pré-condition :** s'authentifier

#### **Séquence principale :**

- 1- lorsque l'utilisateur choisit une connexion dans la liste des connexions, le système déroule le cas d'utilisation englobant
- 2- le système ramène l'utilisateur à la page pour la génération du rapport

#### **Séquence alternative :**

Lorsque la connexion est échouée, le système affiche un message d'erreur et demande à l'utilisateur de se connecter à nouveau.

### **Cas d'utilisation : Gérer les rapports**

**Acteur primaire :** utilisateur

**Pré-condition :** s'authentifier et se connecter à une base de données

#### **Séquence principale :**

- 1- l'utilisateur choisit la table désirée à partir de la base de données
- 2- l'utilisateur passe la requête convenablement
- 3- le système affiche toutes les informations du tables selon la requête
- 4- l'utilisateur génère le rapport

#### **Séquence alternative :**

- 1- l'utilisateur génère le rapport, le système déroule le cas d'utilisation « enregistrer rapport »
- 2- l'utilisateur génère le rapport, le système déroule le cas d'utilisation « télécharger rapport »
- 3- lorsque la requête de l'utilisateur est incorrecte, le système affiche un message d'erreur

### 3.2. Diagramme de séquence « Authentification »

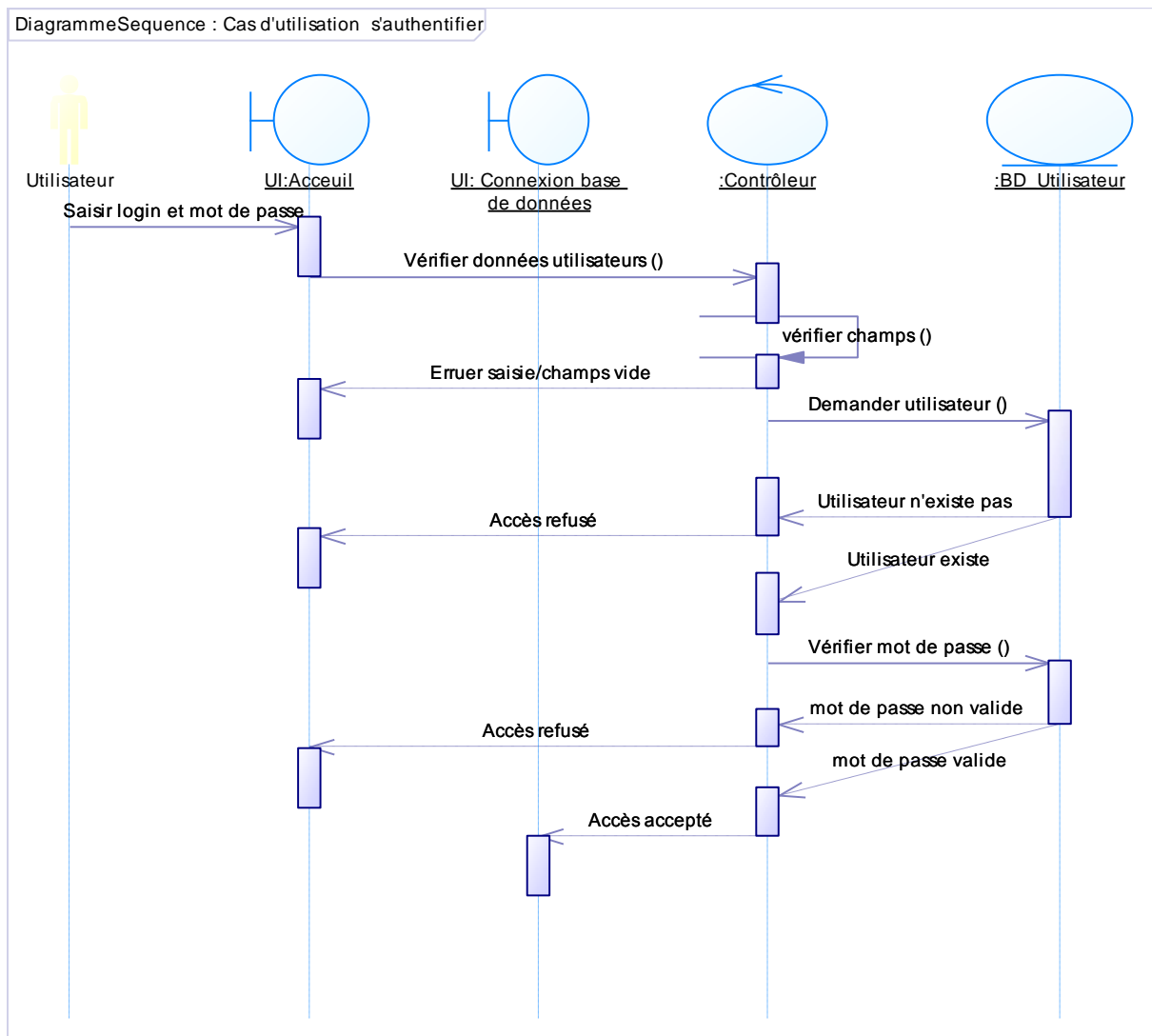
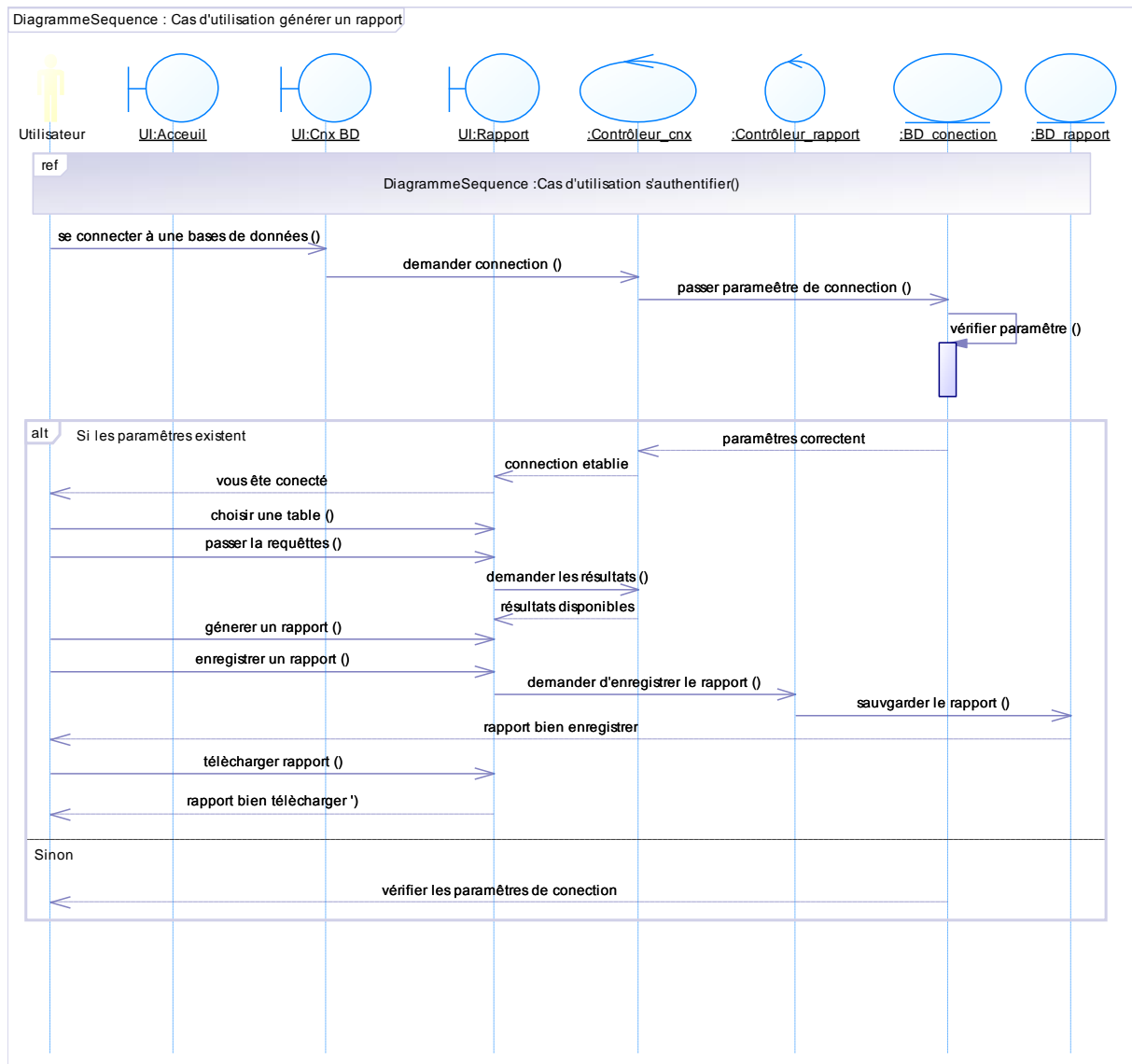


Figure 1.5 : Diagramme de séquence authentification

#### Description :

Afin d'avoir accès à l'application, l'utilisateur doit s'authentifier. Pour ce faire, il saisit le login et le mot de passe à partir de l'interface de connexion. Ces données envoyées directement au contrôleur de connexion seront vérifiées. Si l'utilisateur n'existe pas alors l'accès sera refusé, sinon l'utilisateur sera redirigé vers la page des connexions de bases de données de l'application et pourra réaliser toutes les opérations requises.

### 3.3. Diagramme de séquence « générer des rapports »

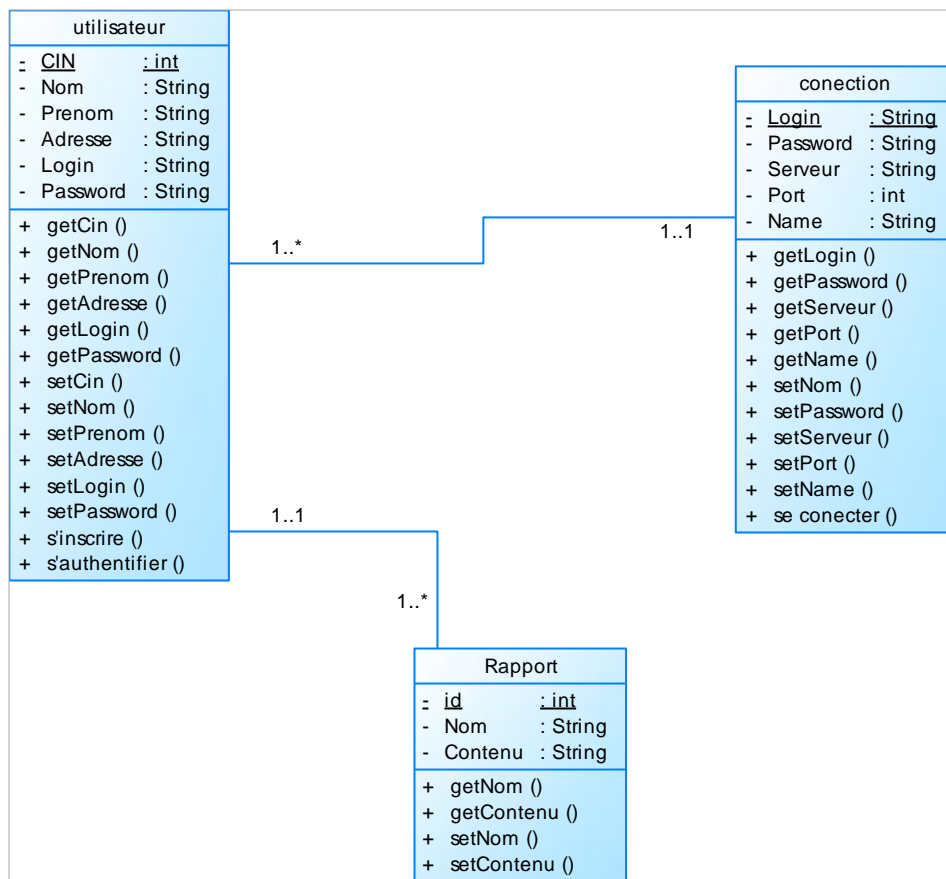


**Figure 1.6 :** Diagramme de séquence générer des rapports

#### Description :

Afin d’avoir accès à l’application, l’utilisateur doit s’authentifier puis se connecte à une base de données désirée et pourra réaliser toutes les opérations requises (générer rapport puis le sauvegarder et/ou le télécharger).

### 3.4. Diagramme de classes



**Figure 1.7 :** Diagramme de classes

#### Description :

Notre diagramme de classes comporte 3 classes :

- ✓ classe utilisateur pour enregistrer les utilisateurs de l'application
- ✓ classe connexion pour manipuler et gérer les connexions aux bases de données
- ✓ classe rapport pour enregistrer les rapports

#### Conclusion

Dans ce chapitre, nous avons spécifié l'application en détaillant les différents besoins fonctionnels du système ainsi que les besoins non fonctionnels. Ensuite, nous avons procédé à une analyse explicite de ces besoins en utilisant le formalisme UML. Enfin, nous avons présenté la conception de cette application. Dans le chapitre suivant, nous allons développer la phase de réalisation dans tous ses détails.

# Chapitre 3 – Réalisation

## Introduction

Dans ce chapitre nous allons présenter en premier lieu l'environnement de travail de coté matériel que logiciel et en deuxième lieu nous allons exposer quelques interfaces de l'application.

### 1. Environnement de travail

Notre environnement de travail est composé d'un environnement matériel et d'un environnement logiciel.

#### 1.1. Environnement matériel

Pour réaliser notre travail, nous avons utilisé comme environnement matériel un ordinateur portable de marque «ASUS» avec un système d'exploitation Windows 7 de 64 bits, ayant un processeur Intel® Core™ i7, doté d'une Ram de 8 Go et d'un disque dur de 1 TO.

#### 1.2. Environnement logiciel

Notre environnement logiciel est composé d'un ensemble de logiciels ainsi qu'un ensemble de librairies.

##### ❖ Logiciels utilisés

- ✓ **Oracle Express Edition 10g** : C'est notre Système de Gestion de la Base de Données (SGBD).
- ✓ **Google Web Toolkit (GWT)** <sup>[7]</sup> : est un ensemble d'outils logiciels développé par Google, permettant de créer et maintenir des applications web dynamique. GWT convertit le code source écrit en java à l'équivalent en JavaScript.
- ✓ **Jasper I Report** : est un outil de Reporting et de statistiques Open Source.
- ✓ **Netbeans** <sup>[8]</sup> : Environnement de Développement Intégré (IDE), placé en open source par Sun en juin 2000. C'est un logiciel regroupant un ensemble d'outils nécessaires au développement des applications dans un langage de programmation.

- ✓ **GlassFish** : est le serveur d'application sur lequel nous avons déployé notre application. Il est présent en natif avec NetBeans.

#### ❖ **Librairies utilisées**

- ✓ **Gwt 2.0.3-gwt-user.jar, Gwt-dev.jar, Gwt-servlet.jar** : sont les bibliothèques nécessaires pour le GWT.
- ✓ **Gxt-2.2.5-gwt17.jar, Gxt-2.2.5-gwt2.jar et Gxt-2.2.5-gwt22.jar** <sup>[9]</sup> : Sencha GXT est le moyen le plus rapide et le plus puissant pour créer de riches applications Web en Java. Sencha GXT, utilisant GWT, est un compilateur qui permet d'écrire des applications en Java et compiler du code hautement optimisé en cross-browser HTML5 et JavaScript.
- ✓ **Jasperreports-3.7.0.jar** : est la bibliothèque de Jasper I Report pouvant être intégrée dans tous types d'applications Java.
- ✓ **JDK 1.6** <sup>[10]</sup> : Le Java Development Kit (JDK) désigne un ensemble de bibliothèques logicielles de base du langage de programmation Java, ainsi que les outils avec lesquels le code Java peut être compilé.

### **1.3. Plate-forme applicative**

Java Enterprise Edition, ou Java EE (anciennement J2EE), est une spécification pour la technique Java de Sun plus particulièrement destinée aux applications d'entreprise.

J2EE est une plate-forme fortement orientée serveur pour le développement et l'exécution d'applications distribuées. Elle est composée de deux parties essentielles :

- un ensemble de spécifications pour une infrastructure dans laquelle s'exécutent les composants écrits en Java : un tel environnement se nomme serveur d'applications.
- un ensemble d'API (Application Programming Interface) qui peuvent être obtenues et utilisées séparément. Pour être utilisées, certaines nécessitent une implémentation de la part d'un fournisseur tiers <sup>[11]</sup>.

## 1.4. Framework <sup>[12]</sup>

JSF, alias "Java Server Faces" est un framework de type MVC (Model Vue Contrôler), destiné aux applications web respectant l'architecture J2EE.

Le premier objectif de JSF, est de procurer un environnement de développement permettant de construire une interface de type web, sans devoir toucher au code HTML et JavaScript. Ceci est réalisé par la mise en place d'un mapping entre l'HTML et les objets concernés. JSF est donc basé sur la notion de composants, comparable à celle de Swing, ou l'état de ces composants est sauvegardé puis restauré au retour de la requête.

Le principal atout de JSF est qu'il tient compte des différentes expériences acquises non seulement avec des technologies de type standard comme les Servlets, les JSP,... mais aussi avec les technologies de type framework comme Struts. Dès lors, JSF propose un framework qui puisse être mis en oeuvre par des outils facilitant le développement d'applications web. Rien n'empêche cependant que JSF puisse être utilisé par codage à la volée, néanmoins, l'utilisation d'un outil approprié est vivement recommandé afin d'exploiter rapidement les possibilités de JSF.

JSF permet :

- une séparation nette entre la couche de présentation et les autres couches
- le mapping HTML/Objet (voir Annexes)
- un modèle riche de composants graphiques réutilisables
- une gestion de l'état de l'interface entre les différentes requêtes
- une liaison simple entre les actions côté client de l'utilisateur et le code Java correspondant côté serveur
- la création de composants customs grâce à une API
- le support de différents clients (HTML, WML, XML, ...) grâce à la séparation des problématiques de construction de l'interface et du rendu de cette interface



## 2. Aperçus & Description

### ❖ Authentification



**Figure 1.8 :** Interface d'authentification

Cette interface permet à l'administrateur de s'authentifier et de se connecter au serveur de la base de données. L'administrateur doit entrer son login et son mot de passe pour accéder à l'application. En cas d'erreur, un message d'alerte s'affiche.

### ❖ Interface d'inscription

A screenshot of a web browser window showing the user registration page. The browser's address bar displays 'localhost:8080/Reporting\_BIAT/faces/inscription\_utilisateur.xhtml'. The page has a light gray background. On the left, there is a form titled 'Inscription' in red. The form contains several input fields: 'CIN', 'Nom', 'Prenom', 'Adresse', 'Login', 'Password', and 'retaper votre Password'. At the bottom of the form are two buttons: 'Enregister' and 'Cliquer ici pour se connecté'.

**Figure 1.9 :** Interface d'inscription d'un utilisateur

Avec cette interface l'utilisateur remplit ces champs afin de s'inscrire.

## ❖ Interface des connections du bases de données

Login	Password	Serveur	Name	Port	
/khawla	khawla	//localhost:	jdbc:mysql:	3306	Se connecter
biat	biat	@localhost	jdbc:oracle:thin	1521	Se connecter

**Figure 1.10 :** Interface de gestion du base de donnée

Lorsque on clique sur bouton se connecter cette interface apparaisse elle permet d'ajouter, de modifier et de supprimer une bases de données et de visualiser la liste des connexions aux bases de données.

## Conclusion

A travers ce chapitre, nous avons présenté clairement la réalisation de l'application en justifiant les choix technologiques et en représentant quelques interfaces graphiques que nous avons jugées les plus importantes.

## **Conclusion Générale**

Au cours de notre stage, nous avons analysé les différents besoins fonctionnels et non fonctionnels, et, en se basant sur la spécification, nous avons élaboré une conception de notre application.

Dans l'étape suivante, nous avons réalisé une application web de reporting.

De point de vue professionnel, ce stage nous a offert l'opportunité d'enrichir notre formation théorique et pratique.

Mais l'acquis le plus important reste l'intégration dans une équipe de travail, de qui nous avons appris à améliorer plusieurs acquis relationnels tels le sérieux, le professionnalisme et l'esprit d'équipe, très utiles pour réussir dans une future vie professionnelle.

Certaines perspectives se dessinent à l'horizon pour l'amélioration de ce travail par l'intégration d'autres fonctionnalités telles que la création d'une application mobile.

## Bibliographie et Webographie

- [1] <https://www.google.fr/url?sa=t&rct=j&q=&esrc=s&source=web&cd=6&cad=rja&ved=0CFEQFjAF&url=http%3A%2F%2Fwww.atuge.org%2Fforum%2Fmedias%2F2013%2Fplaquettes%2Fbiat%2FPresentationBIAT.doc&ei=iKySUr6gGIGJzAOV4ICwAg&usg=AFQjCNHqgGgAmnLgp017y5RVF3Q7LiaaCA> Consulté le « 17/07/2015 »
- [2] Intranet de la BIAT Consulté le « 17/07/2015 »
- [3] <http://www.commentcamarche.net/contents/477-methodes-agiles-rad-xp> Consulté le « 19/07/2015 »
- [4] <http://blog.shaft.fr/?p=333> Consulté le « 19/07/2015 »
- [5] [www.scrumalliance.org](http://www.scrumalliance.org) Consulté le « 20/07/2015 »
- [6] <http://uml.free.fr/> Consulté le « 21/07/2015 »
- [7] <http://doc.ubuntu-fr.org/gwt> Consulté le « 21/07/2015 »
- [8] <http://doc.ubuntu-fr.org/netbeans> Consulté le « 21/07/2015 »
- [9] <http://www.sencha.com/products/gxt/> Consulté le « 21/07/2015 »
- [10] <http://www.enis-androidclub.tk/2013/09/tutoriel-installation-de-lenvironnement.html> Consulté le « 25/07/2015 »
- [11] <http://www.jmdoudoux.fr/java/dej/chap-j2ee-javaee.htm#j2ee-javaee-1> consulté le « 26/08/2015 »
- [12] [http://gardeux-vincent.eu/Documents/ProjetJEE/HPV\\_JSFCastor/jsf.html](http://gardeux-vincent.eu/Documents/ProjetJEE/HPV_JSFCastor/jsf.html) consulté le « 26/08/2015 »