

**UNIVERSITE Sidi Mohamed Ben Abdellah
Faculté Des Sciences Et Techniques Fès**

Département d'Informatique



Projet de fin d'étude

Master Sciences et Techniques

« Systèmes intelligents et réseaux »

LE CONTROLE INTERNE ET LA GESTION DES RISQUES



LIEU DE STAGE : TRESORERIE GENERALE DU ROYAUME, RABAT

Réalisé par :

M. YOUSSEF AFRACHE

M. ABDELILAH HAMIDACHE

Soutenu le 19/06/2014

Encadré par :

Mr. LAHCEN BOUHACHEMOUD

Mme. LOUBNA LAMRINI

Devant le jury composé de :

Pr. Med Chaouki abounaima

Pr. Loubna Lamrini

Pr. Said Najah

Pr. Khalid Abbad

Année Universitaire 2013-2014





Dédicaces

Dédicaces

Nous dédions ce modeste travail.

A nos familles avec tous nos sentiments de respect, d'amour, de gratitude et de reconnaissance pour tous les sacrifices déployés pour nous élever dignement et assurer notre éducation dans les meilleurs conditions.

Nos pères qui n'ont rien épargné pour nous satisfaire.

Nos chères mères, la source inépuisable de notre bonheur, les êtres qui ne cessent de nous prodiguer amour et douceur.

A nos professeurs sans exception, pour leurs efforts afin de nous assurer une formation bien solide.

A nos amis sans exception.



Remerciements

Remerciements

Nous tenons tout d'abord à présenter nos sincères remerciements à notre encadrant **Madame Loubna LAMRINI** qui nous a permis à travers son suivi rigoureux, son entière disponibilité et les conseils pertinents qu'elle nous a prodigués tout au long de ce stage de mieux cadrer et appréhender les différentes étapes de notre projet de fin d'étude.

Nous remercions également **M. Aziz BOUAZZAoui**, Directeur des ressources et du système d'information qui a eu la bienveillance d'accepter de nous accueillir en tant que stagiaires à la Trésorerie Générale du Royaume. A ce titre, nous tenons à saluer la qualité de l'accueil et les conditions de travail encourageantes qui nous ont été offertes tout au long des quatre mois de stage dans cette prestigieuse institution.

Nos remerciements sont ensuite vivement adressés à **Monsieur Lahcen BOUHCHAMOUD**, Chef du service du développement à la Direction des ressources et du système d'information. Ses remarques et suggestions ont fortement contribué à l'amélioration de la qualité de ce rapport de fin d'étude. Nous exprimons aussi notre gratitude à l'ensemble des membres de son équipe d'assistance pour la gentillesse et la patience qu'ils ont manifestées à notre égard.

Nous tenons à remercier aussi toute **l'équipe pédagogique** de la faculté des sciences et techniques de FES, surtout les intervenants professionnels responsable de notre filière « systèmes intelligents et réseaux » et sans oublier **les membres du jury** qui ont accepté d'évaluer notre travail et de l'enrichir par leurs propositions.



Résumé

Résumé

Dans le cadre de notre projet de fin d'étude, les services de la Trésorerie Générale du Royaume ont bien voulu nous accepter en tant que stagiaires. C'est une opportunité de marque que de réaliser un stage pratique dans l'une des administrations les plus importantes du ministère de l'Economie et des Finances, et qui se situe au centre d'un maillage institutionnel constitué d'administrations publiques, d'établissements publics, de collectivités locales et d'autres grandes institutions financières, tous concernés par la gestion des deniers publics.

Pour la réalisation de ce projet, nous avons centré notre travail sur la conception et le développement d'une application web pour le contrôle interne et la gestion des risques et qui est basée sur l'architecture J2EE avec les framework JPA /HIBERNATE et SPRING. Cette application comprend d'une part, une base de données relative à l'archivage des travaux et d'autre part, une interface conviviale et simple dédiée à sa gestion.

Pour sa conception et son développement, nous nous sommes conformés à un cahier des charges délimitant le processus de gestion des risques à la Trésorerie Générale du Royaume. Ce processus comprend l'identification, l'évaluation et le traitement des risques identifiés.



SOMMAIRE

La liste des figures	8
I. INTRODUCTION	9
II. CHAPITRE 1 : CONTEXTE DU PROJET	10
1. PRESENTATION DE L'ORGANISME D'ACCUEIL.....	10
2. OBJECTIF DU PROJET	13
3. CAHIER DES CHARGES.....	13
4. Conduite du projet	16
III. CHAPITRE 2 : ETUDE FONCTIONNELLE ET TECHNIQUE	17
1. LA DEMARCHE : LE PROCESSUS EN Y.....	17
2. ETUDE FONCTIONNELLE	18
2.1. Capture des besoins fonctionnels	18
2.2. Diagramme de cas d'utilisation	23
2.3. Diagramme de séquence.....	29
2.4. Diagramme de classe.....	35
3. ETUDE TECHNIQUE	36
3.1. Architecture physique	36
3.2. Architecture logique	37
IV. CHAPITRE 3 : MISE EN ŒUVRE DU PROJET	39
1. REALISATION.....	39
1.1. Environnement de développement Eclipse	39
1.2. Serveur web et d'application : Apache Tomcat.....	39
1.3. SGBD MySQL :.....	39
1.4. Architecture J2EE :.....	40
1.5. Framework de persistance : Hibernate	40
1.6. Spring MVC :.....	40
1.7. Framework de présentation : JPA	40
1.8. WampServer :.....	41
1.9. Entreprise Architect :	41
1.10. Spring Security :.....	41
2. PRESENTATION DE L'APPLICATION	42
2.1. Formulaire d'authentification	42



2.2.	Interface d'ajout d'un risque	44
2.3.	Fiche des risques	46
2.4.	Suppression d'un risque	47
2.5.	Interface d'ajout d'un contrôle	48
2.6.	Interface d'ajout d'une procédure	48
2.7.	L'envoi de demande d'évaluation	50
2.8.	Traitement d'évaluation	51
2.9.	Suivi d'action	53
2.10.	Création de la fiche de reporting.....	54
V.	CONCLUSION ET PERSPECTIVES	55
	BIBLIOGRAPHIE	56
	WEBOGRAPHIE	57

La liste des figures

Figure 1 : Organigramme de la trésorerie générale du royaume.....	12
Figure 2 : Diagramme de gant.....	16
Figure 3 : Processus 2TUP (en Y)	18
Figure 4 : Diagramme de cas d'utilisation du service en charge de l'analyse des risques	24
Figure 5 : Diagramme de cas d'utilisation du service du contrôle interne.....	25
Figure 6 : Diagramme de cas d'utilisation du responsable du contrôle interne	26
Figure 7 : Diagramme de cas d'utilisation du relai local.....	27
Figure 8 : Diagramme de cas d'utilisation de l'administrateur	28
Figure 9 : Diagramme de séquence d'authentification	30
Figure 10 : Diagramme de séquence d'ajout d'un risque	32
Figure 11 : Diagramme de séquence d'ajout d'un utilisateur.....	34
Figure 12: Diagramme de classe.....	35
Figure 13 : Architecture physique du projet	37
Figure 14 :L'architecture en couche de notre application.....	38
Figure 15 : Page d'authentification.....	42
Figure 16 : Echec d'authentification.....	43
Figure 17 : Service en charge de l'analyse des risques	43
Figure 18 : Service du contrôle interne	44
Figure 19 : Responsable du contrôle.....	44
Figure 20 : Créer un risque	45
Figure 21 : Erreur d'ajout d'un risque.....	46
Figure 22 : risque ajouté avec succès	46
Figure 23 : La fiche des risques	47
Figure 24 : Supprimer un risque.....	47
Figure 25 : Fiche après suppression.....	48
Figure 26 : Ajout d'un contrôle	48
Figure 27 : Définir une procédure	49
Figure 28 : La fiche des procédures.....	49
Figure 29 : Envoyer une demande d'évaluation.....	50
Figure 30 : la liste des risques à évaluer	50
Figure 31 : Evaluer risque	50
Figure 32 : Valider l'évaluation.....	51
Figure 33 : Les risques à traiter	51
Figure 34 : Evaluation en cours de traitement	52
Figure 35 : Traiter évaluation	52
Figure 36 : Annuler une évaluation	52
Figure 37 : Détail d'une évaluation.....	53
Figure 38 : Effectuer un suivi.....	53
Figure 39 : Affichage d'un suivi.....	53
Figure 40 : Création de la fiche de reporting.....	54
Figure 41 : Affichage de la fiche de reporting.....	54



I. INTRODUCTION

Dans le cadre de notre apprentissage à la faculté des sciences et techniques de Fès, nous sommes hautement encouragés à accompagner notre cursus de formation théorique par un stage pratique permettant de nous familiariser avec le monde du travail tout en perfectionnant nos connaissances dans notre domaine d'étude.

Pour ce faire, nous avons opté pour un stage à la Trésorerie Générale du Royaume (TGR). Ce choix répond à une considération majeure qui consiste à travers la thématique traitée, à contribuer modestement aux efforts entrepris par l'institution pour la mise en place d'applicatifs destinés à renforcer son dispositif de contrôle interne, lequel est adossé à une démarche de gestion des risques.

Dans ce cadre, le travail qui nous a été confié s'inscrit parfaitement dans cette optique. En effet, notre projet a porté sur la conception et la réalisation d'une application dédiée au contrôle interne et à la gestion des risques.

Notre rapport de stage est ainsi structuré en trois parties. Dans la première partie, nous allons présenter le contexte général dans lequel s'est déroulé le projet. A cet effet, un premier axe sera réservé à la présentation de l'organisme d'accueil et un deuxième axe concernera l'objectif du projet ainsi que le cahier des charges.

La deuxième partie est réservée à la présentation de la démarche de développement suivie de la conduite de ce projet ainsi que l'étude fonctionnelle et technique réalisée.

La dernière partie, traitera la mise en œuvre du projet. Dans ce cadre, nous allons présenter les technologies choisies et la partie application qui donnera un aperçu de l'application et son apparence finale.



II. CHAPITRE 1 : CONTEXTE DU PROJET

Ce chapitre présente le contexte général dans lequel s'est déroulé le stage. Il présente dans une première phase l'organisme d'accueil et dans une seconde phase l'objectif du projet ainsi que le cahier des charges.

1. PRESENTATION DE L'ORGANISME D'ACCUEIL

Nous avons effectué notre stage au sein de la Trésorerie Générale du Royaume. Celle-ci est considérée comme étant l'une des administrations les plus importantes du ministère de l'Economie et des Finances, en ce sens que l'ensemble des flux financiers et comptables de l'Etat et des collectivités locales transite à travers ses circuits.

Elle est également au centre d'un maillage institutionnel constitué d'administrations publiques, d'établissements publics, de collectivités locales et d'autres grandes institutions financières, tous concernés par la gestion des deniers publics.

Missions et attributions

✓ **Le recouvrement des créances publiques**

La TGR assure, par le biais de son vaste réseau de comptables publics, la perception des recettes fiscales et non fiscales, à travers notamment :

- la gestion du contentieux administratif et judiciaire relatif au recouvrement et l'assistance des percepteurs en la matière.
- la prise en charge des ordres de recettes au titre du budget général de l'Etat, des budgets SEGMA et des comptes spéciaux du Trésor.
- la centralisation des prises en charges et des recouvrements au titre des amendes et condamnations pécuniaires.
- la gestion des comptes de prêts et d'avances accordées par le Trésor et de «fonds de roulement» consentis par des organismes de financement des projets publics.
- l'élaboration des statistiques concernant la situation du recouvrement de créances publiques.



✓ **Le contrôle et le paiement des dépenses publiques**

La TGR assure le contrôle et le règlement des dépenses publiques. Ainsi, le réseau de la TGR est chargé de contrôler la régularité des engagements de la quasi-totalité des dépenses de l'Etat. Elle assure à travers son réseau de comptables, le règlement desdites dépenses. En effet, au vu des propositions d'engagement et des ordres de paiement transmis par les ordonnateurs accrédités, les services de la TGR procèdent au règlement des créances de l'Etat.

La Trésorerie Générale assure également par le biais du Centre National des Traitements (CNT), le contrôle et le traitement de la paie de près de 650.000 fonctionnaires.

✓ **La gestion des finances locales**

A travers son réseau de trésoriers et receveurs communaux, la TGR assure la gestion des budgets de 1659 collectivités locales, de 86 groupements et de 41 arrondissements. En effet, la TGR procède au recouvrement de leurs créances, au règlement de leurs dépenses et à la paie de leur personnel.

La TGR met à contribution également son expertise en offrant le conseil et l'assistance nécessaire aux collectivités locales. Ce conseil qui est de nature juridique et financière, concerne, entre autres, la modernisation des procédures comptables, l'analyse financière et l'élaboration des tableaux de bord.

✓ **La gestion des dépôts au Trésor**

La TGR assure la mission de gestion des dépôts au Trésor. Elle participe à travers cette activité au financement de la trésorerie de l'Etat. A ce titre, elle gère les comptes des entreprises et établissements publics qui sont soumis à l'obligation de dépôt de leurs fonds au Trésor. Cette activité est étendue également à la gestion des dépôts des autres personnes morales ou privées.

✓ **La production de l'information financière et comptable**

La TGR assure la centralisation des opérations comptables de l'Etat et des collectivités locales, et de ce fait, elle constitue une référence en matière de production et de valorisation de l'information comptable de l'Etat et des collectivités locales.

La production de l'information comptable permet ainsi de :

- décrire précisément les opérations budgétaires et financières.
- restituer rapidement une information fiable et indispensable à la prise de décision.
- préparer les documents relatifs à la reddition des comptes.

L'organigramme de la Trésorerie Générale du Royaume est structuré comme suit:

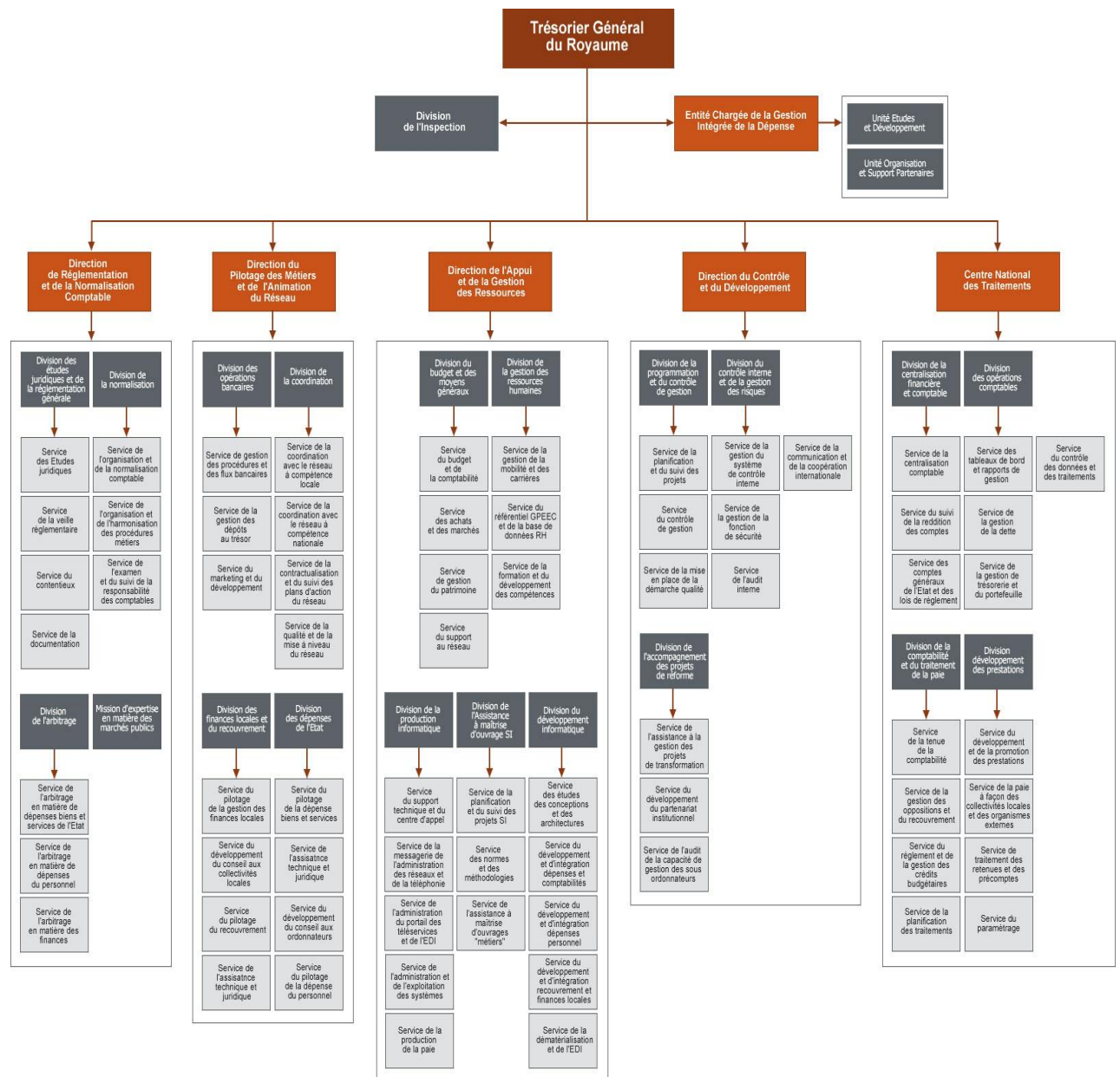


Figure 1 : Organigramme de la trésorerie générale du royaume



2. OBJECTIF DU PROJET

Compte tenu de l'étendue de son périmètre de compétence et de l'importance de son rôle dans les finances publiques, la Trésorerie Générale du Royaume a mis en place un dispositif de contrôle interne adossé à une démarche de gestion des risques.

Cette démarche est motivée par une volonté d'améliorer l'exercice des métiers, de s'assurer du respect des textes qui les régissent et de promouvoir la qualité de l'information produite. Elle s'avère, par ailleurs, parfaitement alignée par rapport aux grandes tendances qui se profilent en matière de gestion des finances publiques à savoir : la gestion par la performance et la promotion de la transparence.

L'objectif que nous nous sommes assigné durant ce stage consiste en la conception et la création d'une application pour le contrôle interne et la gestion des risques. Celle-ci comprend:

- Une base de données pour l'archivage de ces travaux ;
- Une interface conviviale et simple qui permet la gestion de celle-ci.

3. CAHIER DES CHARGES

Notre projet de fin d'étude est dédié à la réalisation d'une application web pour le contrôle interne et la gestion des risques.

Gestion des risques à la TGR

D'une manière schématique, le processus de gestion des risques comprend plusieurs phases :

Phase 1 : L'identification des risques

L'identification des risques est le résultat d'une analyse introspective qui porte sur l'organisation, sur son activité et sur son environnement. C'est un exercice qui consiste à relever les événements, les situations ou les actions qui tendent à empêcher l'institution ou l'une de ses entités à atteindre ses objectifs.



Phase 2 : L'évaluation des risques

L'évaluation des risques a pour objectif de quantifier les risques et de mesurer, d'une certaine manière, leur importance. Cela a pour but de :

- les hiérarchiser les uns par rapport aux autres;
- suivre leur évolution dans le temps.

A ce niveau, la cartographie des risques est une représentation graphique de l'importance des risques selon deux dimensions : la fréquence et l'impact.

Phase 3 : le traitement des risques ou le contrôle interne

Globalement, la réponse aux risques consiste à mettre en œuvre un dispositif de contrôle interne au sein de l'organisation. Cependant, la réponse à un risque, pris individuellement, consiste à l'analyser pour arrêter la stratégie à adopter pour y faire face. Ceci revient à choisir entre l'une des quatre options possibles à savoir :

- accepter le risque ;
- l'éviter ;
- le partager ou ;
- le réduire.

Lorsque l'option retenue est de réduire le risque, il s'agit à ce moment de l'analyser pour identifier les facteurs du risque et définir les activités de contrôle et les actions nécessaires pour le couvrir.

Notre objectif est de permettre aux utilisateurs de bien gérer les risques, les contrôles, les actions que nous détaillerons dans ce cahier des charges.

- ✓ **Utilisateurs** : se constitue d'un administrateur, service en charge de l'analyse des risques, service du contrôle interne, responsable du contrôle et relai local.
- ✓ **Administrateur** : peut gérer les utilisateurs du système.
- ✓ **Service en charge de l'analyse des risques** :



- Gestion des risques (ajouter, modifier, consulter, supprimer risques).
- Envoyer des demandes d'évaluations.
- Traitement des évaluations.
- Elaborer la cartographie des risques.

✓ **Service du contrôle interne :**

- Définir les actions.
- Traiter les fiches du reporting.
- Définir les contrôles.
- Suivre les actions.
- Valider les plans de test.
- Apprécier les fiches diagnostiques.

✓ **Responsable du contrôle :**

- Gérer les ressources.
- Evaluer les risques.
- Faire le suivi des actions.
- Visualiser les plans de test et élaborer la fiche de reporting qui est constituée de la fiche du contrôle, du risque et celle d'action.

✓ **Relai local :**

- Elaborer le plan de test.
- Réaliser les diagnostics des procédures.
- Réaliser les tests comptables.

4. Conduite du projet

Dans le cadre de la conduite du projet, la réalisation d'un planning à suivre tout au long du stage de fin d'études s'impose. Ainsi, le stage a débuté le 04 février 2014. Du coup, une réunion a été tenue afin de définir le calendrier du projet.

Le planning sur lequel on s'est mis d'accord est subdivisé en quatre grandes étapes:

- ✓ La première est l'étape de documentation et d'expression des besoins.
- ✓ La seconde est l'étape pour étudier les frameworks spring et hibernate.
- ✓ La troisième étape, quant à elle, traite de la conception.
- ✓ La dernière étape est consacrée à l'implémentation, le jeu de tests et le déploiement.

Le planning des étapes de déroulement du projet est présenté par la figure ci-dessous qui représente le diagramme de GANTT qui permet de rendre plus simple le suivi d'avancement du projet.

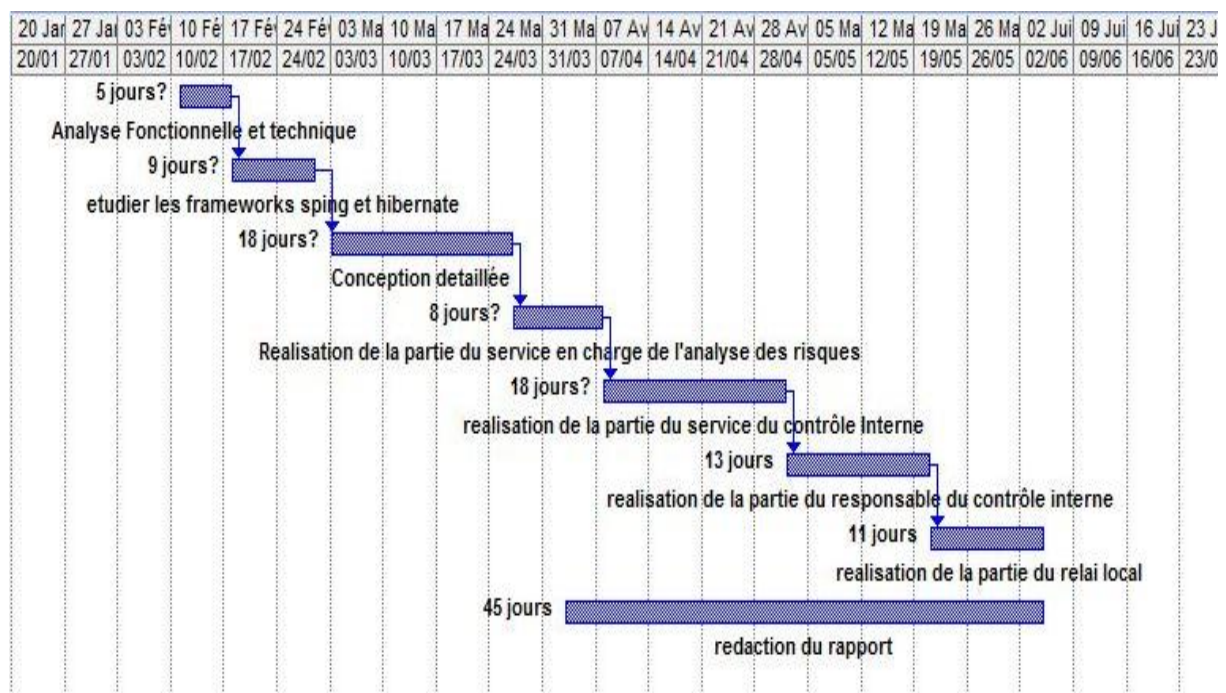


Figure 2 : Diagramme de gant

III. CHAPITRE 2 : ETUDE FONCTIONNELLE ET TECHNIQUE

Ce chapitre présente la démarche de développement suivie de la conduite de ce projet ainsi que l'étude fonctionnelle et technique réalisée.

1. LA DEMARCHE : LE PROCESSUS EN Y

Le processus 2TUP est un processus unifié. Il gère la complexité technologique en réservant une part à la technologie dans son processus de développement.

Le processus 2TUP propose un cycle de développement qui sépare les aspects techniques des aspects fonctionnels et propose une étude parallèle des deux branches : fonctionnelle (étude de l'application) et la technique (étude de l'implémentation).

Le processus 2TUP s'articule autour de trois branches :

- Une branche technique.
- Une branche fonctionnelle.
- Une branche de conception et de réalisation.

Les principales étapes qu'on a utilisées dans notre projet se présentent comme suit :

- ✓ **Capture des besoins fonctionnels** : cette phase a pour objectif de définir :
 - La frontière fonctionnelle entre le système et son environnement.
 - Les activités attendues des différents utilisateurs par rapport au système.
- ✓ **Analyse** : Consiste à étudier précisément les spécifications fonctionnelles de manière à obtenir une idée de ce que va réaliser le système en terme de métier.
- ✓ **Conception générique** : Définit les composants nécessaires à la construction de l'architecture technique. Cette conception est

complètement indépendante des aspects fonctionnels.

Elle permet de générer le modèle de conception technique qui définit les Frameworks.

- ✓ **Codage** : Permet d'effectuer la production des composants et les tests des unités de code au fur et à mesure de leur réalisation.

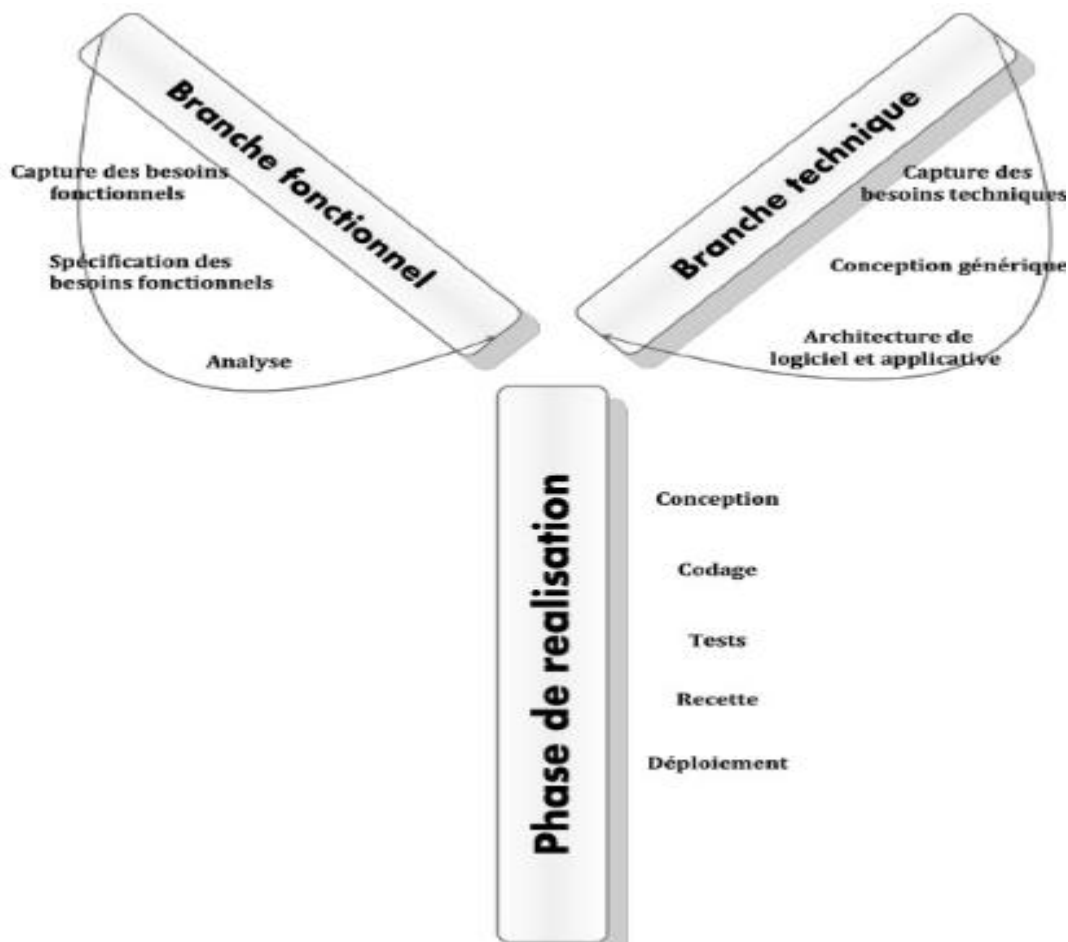


Figure 3 : Processus 2TUP (en Y)

2. ETUDE FONCTIONNELLE

2.1. Capture des besoins fonctionnels

2.1.1. Identification des acteurs

Les acteurs sont des entités externes qui interagissent (saisie de données, affichage d'informations, ...) avec le système. Ces acteurs permettent de cerner l'interface que le système va devoir offrir à son environnement.



Dans notre système, on distingue deux acteurs:

a- **L'utilisateur** qui peut appartenir à l'un des profils suivants :

- Service en charge de l'analyse des risques.
- Service du contrôle interne.
- Responsable du contrôle.
- Relai local.

b- **L'administrateur** qui peut gérer les utilisateurs.

- Le rôle du service en charge de l'analyse des risques:
 - Gestion des risques (ajouter, modifier, consulter, supprimer des risques).
 - Envoyer des demandes d'évaluation.
 - Traitement d'évaluation.
 - Elaboration de la cartographie des risques.
- Le rôle du service du contrôle interne :
 - Gestion des contrôles (ajouter, modifier, consulter, supprimer des contrôles).
 - Définir les actions.
 - Suivi des actions.
 - Validation du plan de test.
 - Traiter les fiches reporting.
 - Appréciation fiche diagnostique.
- Le rôle du responsable du contrôle :
 - Elaborer fiche reporting (Proposer (risque, contrôle)).
 - Evaluer les risques.
 - Visualiser le plan de test.
- Le rôle du relai local :
 - Elaborer le plan de test.
 - Réaliser les fiches diagnostiques (Proposer (risque, contrôle)).
 - Réaliser les tests comptables.
- Le rôle de l'administrateur :
 - Ajouter, modifier, supprimer, consulter les utilisateurs.

2.1.2. Identification des cas d'utilisation

Liste des cas d'utilisation :

A partir du cahier des charges, nous établirons les cas suivants :

Pour chaque cas d'utilisation nous précisons les acteurs, les rôles et les messages échangés.

Cas d'utilisation	Acteur	Rôle	Description des cas d'utilisation
Gestion des risques	Utilisateur	Service en charge de l'analyse des risques	L'utilisateur peut ajouter, modifier, supprimer et consulter les risques
Envoyer des demandes d'évaluations	Utilisateur	Service en charge de l'analyse des risques	L'utilisateur envoi des demandes d'évaluations des risques au responsable du contrôle
Traitement d'évaluation	Utilisateur	Service en charge de l'analyse des risques	L'utilisateur traite les risques évalué par les responsables du contrôle
Elaboration de la cartographie des risques	Utilisateur	Service en charge de l'analyse des risques	L'utilisateur affiche l'importance des risques
Gestion des contrôles	Utilisateur	Service du contrôle interne	L'utilisateur peut ajouter, modifier, supprimer et consulter les

			contrôles
Définir les actions	Utilisateur	Service du contrôle interne	L'utilisateur peut créer des actions
Suivi des actions	Utilisateur	Service du contrôle interne	L'utilisateur peut voir l'état d'avancement d'une action
Validation du plan de test	Utilisateur	Service du contrôle interne	L'utilisateur peut valider les tests comptables à réaliser pour le relai local
Traiter les fiches reporting	Utilisateur	Service du contrôle interne	L'utilisateur traite la fiche de reporting pour la mise à jour du référentiel du contrôle
Appréciation fiche diagnostique	Utilisateur	Service du contrôle interne	L'utilisateur traite la fiche diagnostique pour la mise à jour du référentiel du contrôle
Elaborer fiche reporting	Utilisateur	Responsable du contrôle	L'utilisateur rend compte de la mise en œuvre du contrôle interne et propose une amélioration du dispositif du contrôle

Evaluer les risques	Utilisateur	Responsable du contrôle	L'utilisateur évalue les risques envoyé par le service en charge de l'analyse des risques
Visualiser le plan de test	Utilisateur	Responsable du contrôle	L'utilisateur peut consulter le plan de test
Elaborer le plan de test	Utilisateur	Relai local	L'utilisateur propose les tests comptables à réaliser et les envoyer au service du contrôle interne
Réaliser les fiches diagnostiques	Utilisateur	Relai local	L'utilisateur peut examiner les procédures pour évaluer la pertinence du contrôle et proposer de nouveau (risques, contrôles)
Réaliser les tests comptables	Utilisateur	Relai local	L'utilisateur peut réaliser les tests comptables validés par le service du contrôle interne
Gestion	Administrateur		L'administrateur

d'utilisateurs			peut ajouter, modifier, supprimer, consulter les utilisateurs
Authentification	Administrateur ou utilisateur		L'administrateur et les utilisateurs peuvent s'authentifier

2.2. Diagramme de cas d'utilisation

Les diagrammes de cas d'utilisation sont des diagrammes UML utilisés pour donner une vision globale du comportement fonctionnel d'un système logiciel. Ils sont utiles pour assurer des présentations auprès de la direction ou des acteurs d'un projet. Un cas d'utilisation représente une unité discrète d'interaction entre un utilisateur et un système et se considère comme une unité significative de travail. Dans un diagramme de cas d'utilisation, les utilisateurs sont appelés acteurs, et interagissent avec les cas d'utilisations.

Après l'étude de chaque cas d'utilisation, nous avons élaboré le diagramme des cas d'utilisation de chaque rôle :

2.2.1. Service en charge de l'analyse des risques

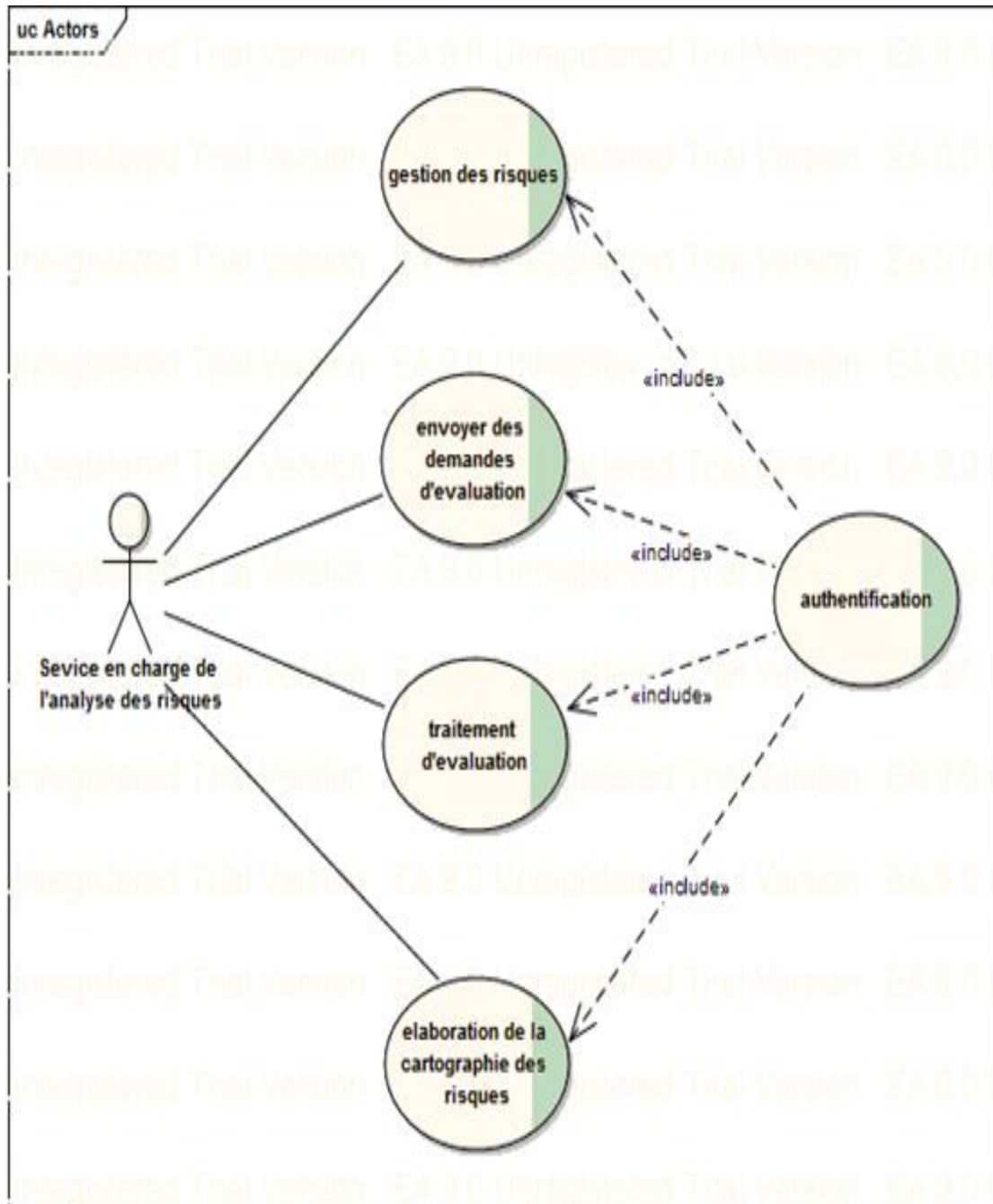


Figure 4 : Diagramme de cas d'utilisation du service en charge de l'analyse des risques

2.2.2. Service du contrôle interne

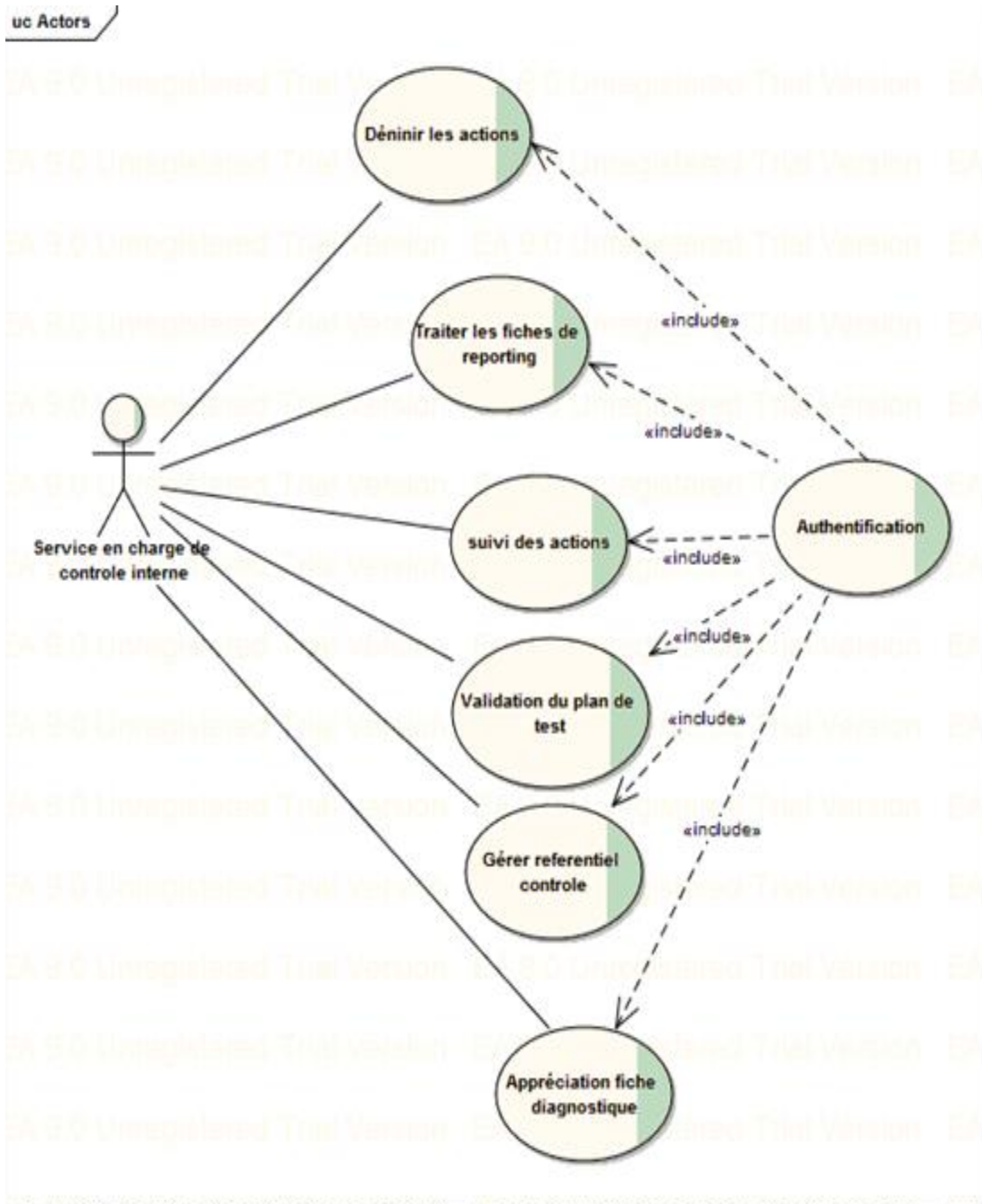


Figure 5 : Diagramme de cas d'utilisation du service du contrôle interne

2.2.3. Responsable du contrôle interne

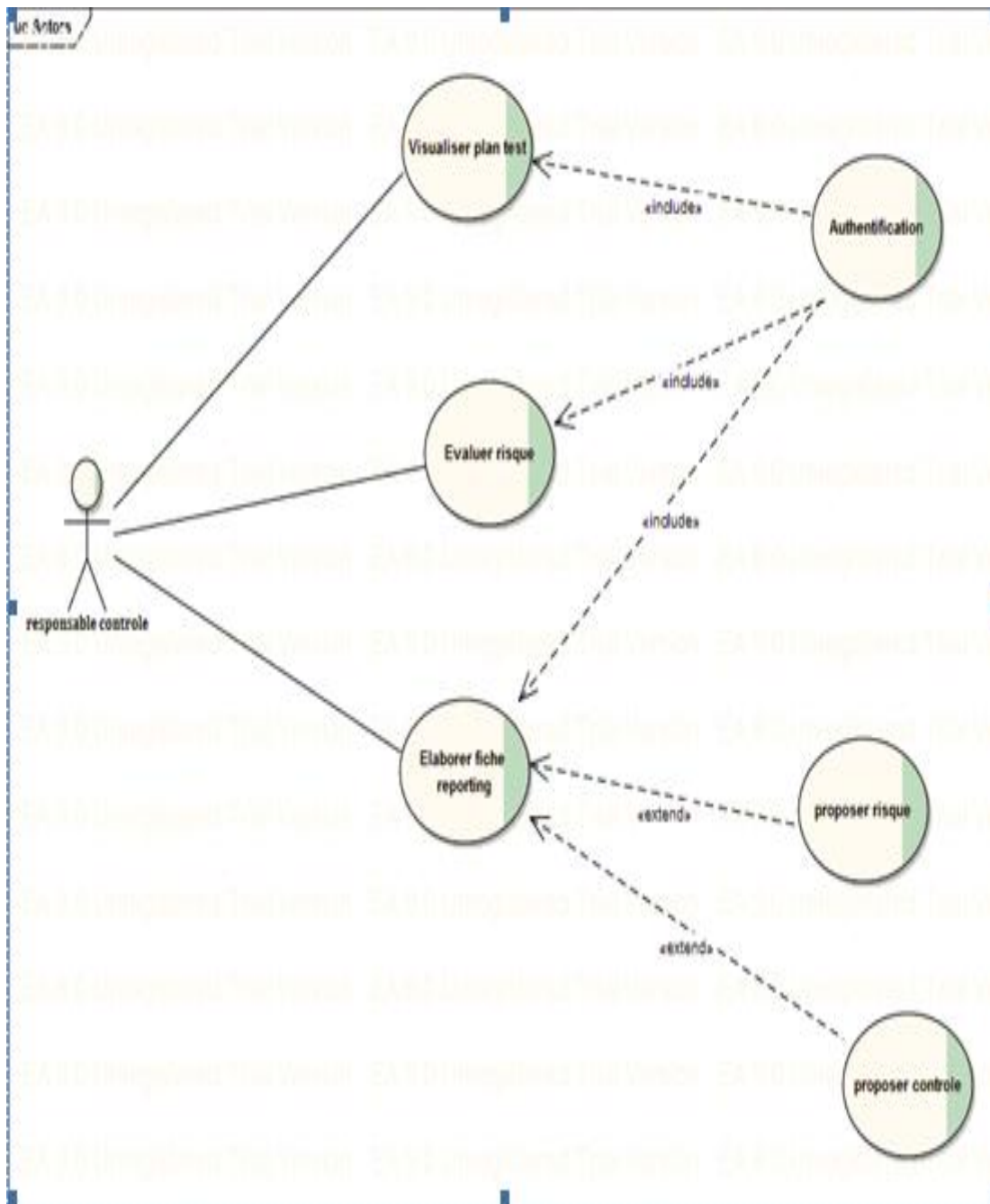


Figure 6 : Diagramme de cas d'utilisation du responsable du contrôle interne

2.2.4. Relai local

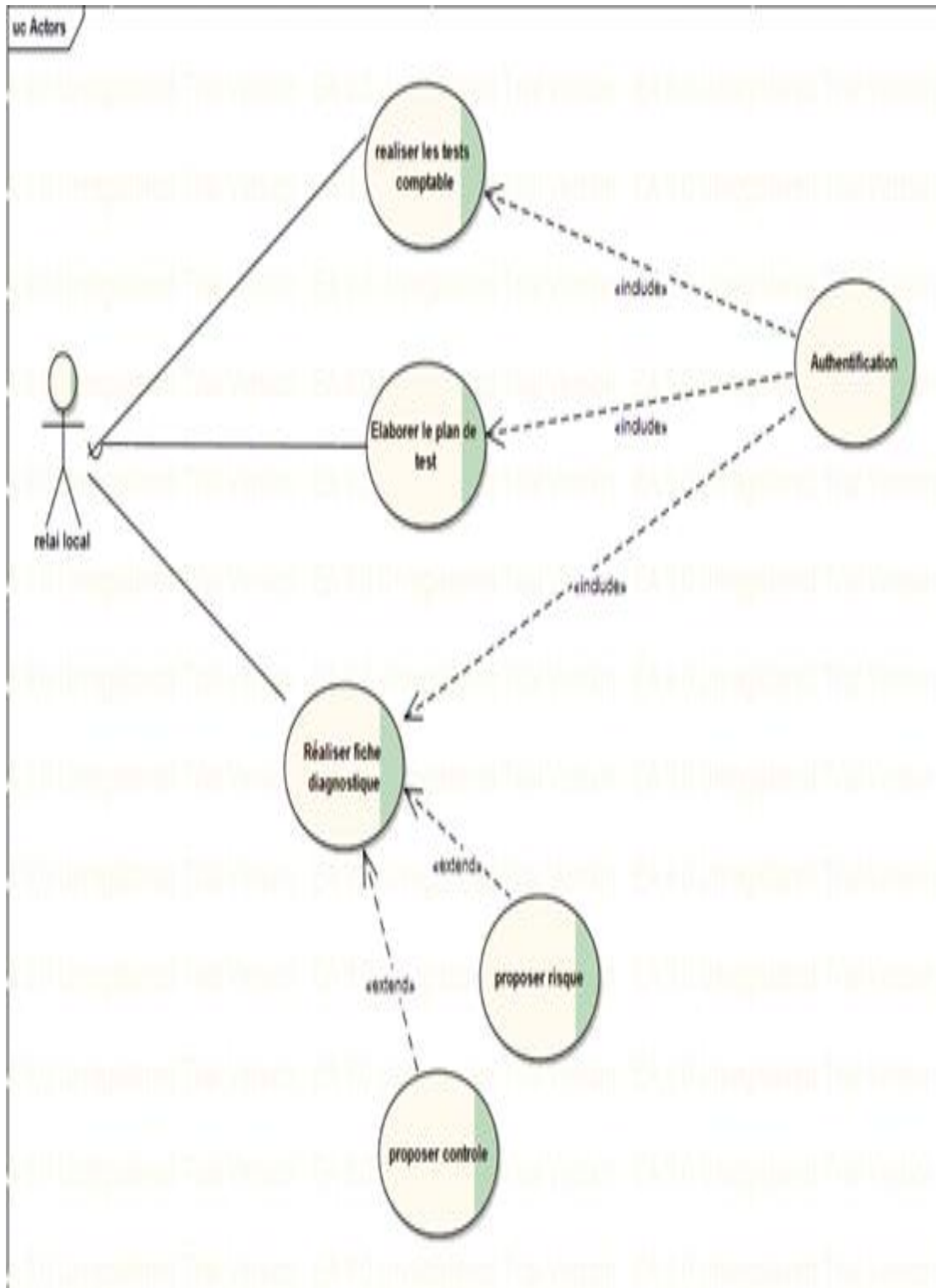


Figure 7 : Diagramme de cas d'utilisation du relai local

2.2.5. Administrateur

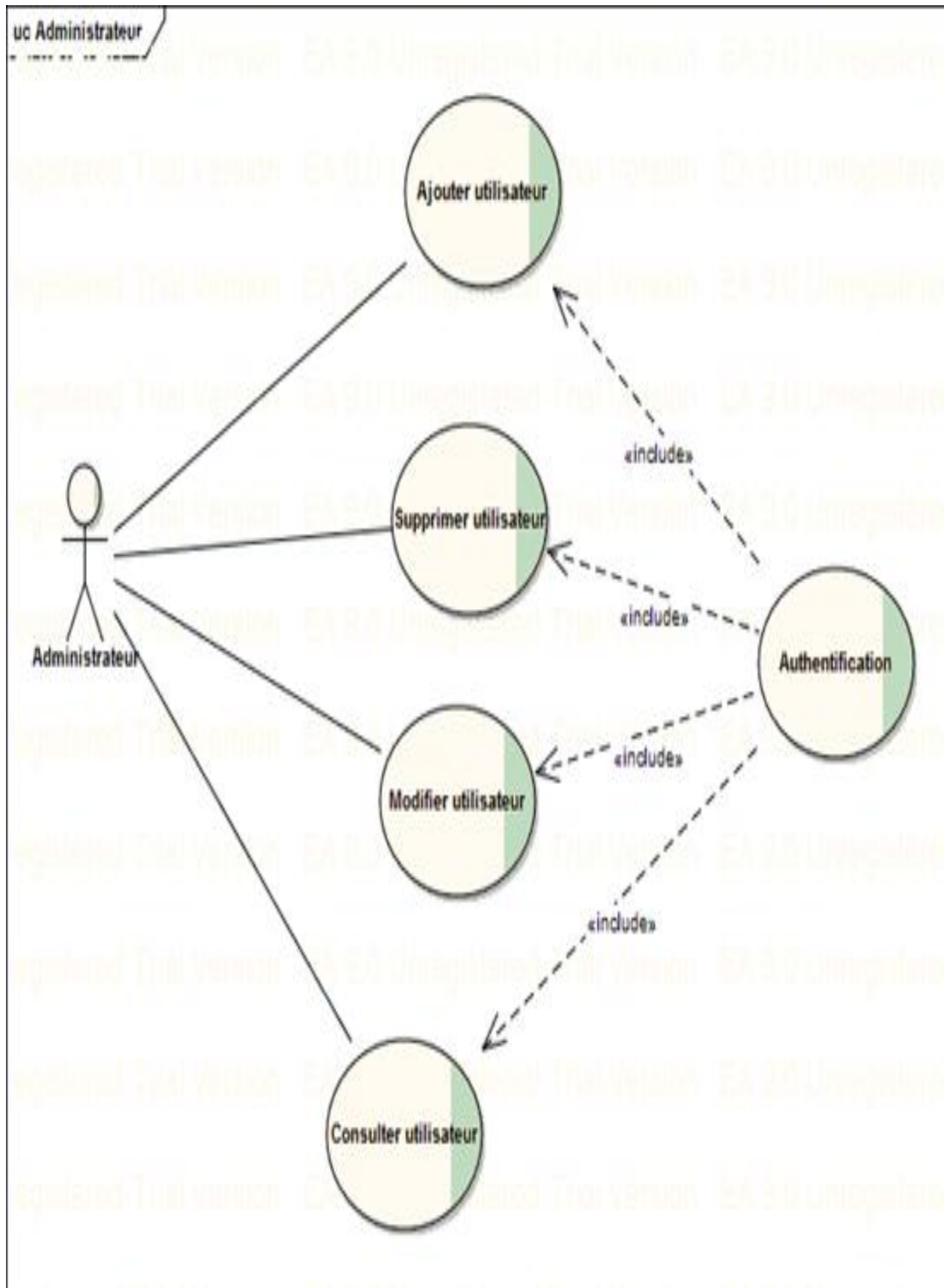


Figure 8 : Diagramme de cas d'utilisation de l'administrateur

2.3. Diagramme de séquence

Le diagramme de séquence est la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique dans la formulation Unified Modeling Language.

Diagramme de séquence d'authentification :

- Acteur principal : Administrateur ou utilisateur.
- Intention de l'acteur : l'utilisateur veut s'authentifier.

Pour le cas d'utilisation d'authentification, on distingue les scénarios suivants :

- Post condition : l'utilisateur quitte la fenêtre d'authentification.
- Scénario nominal : l'utilisateur saisie le login et le mot de passe et clique sur le bouton réservé à la connexion.
- Scénario d'erreur: l'utilisateur n'a rien saisi. Login ou mot de passe incorrecte.

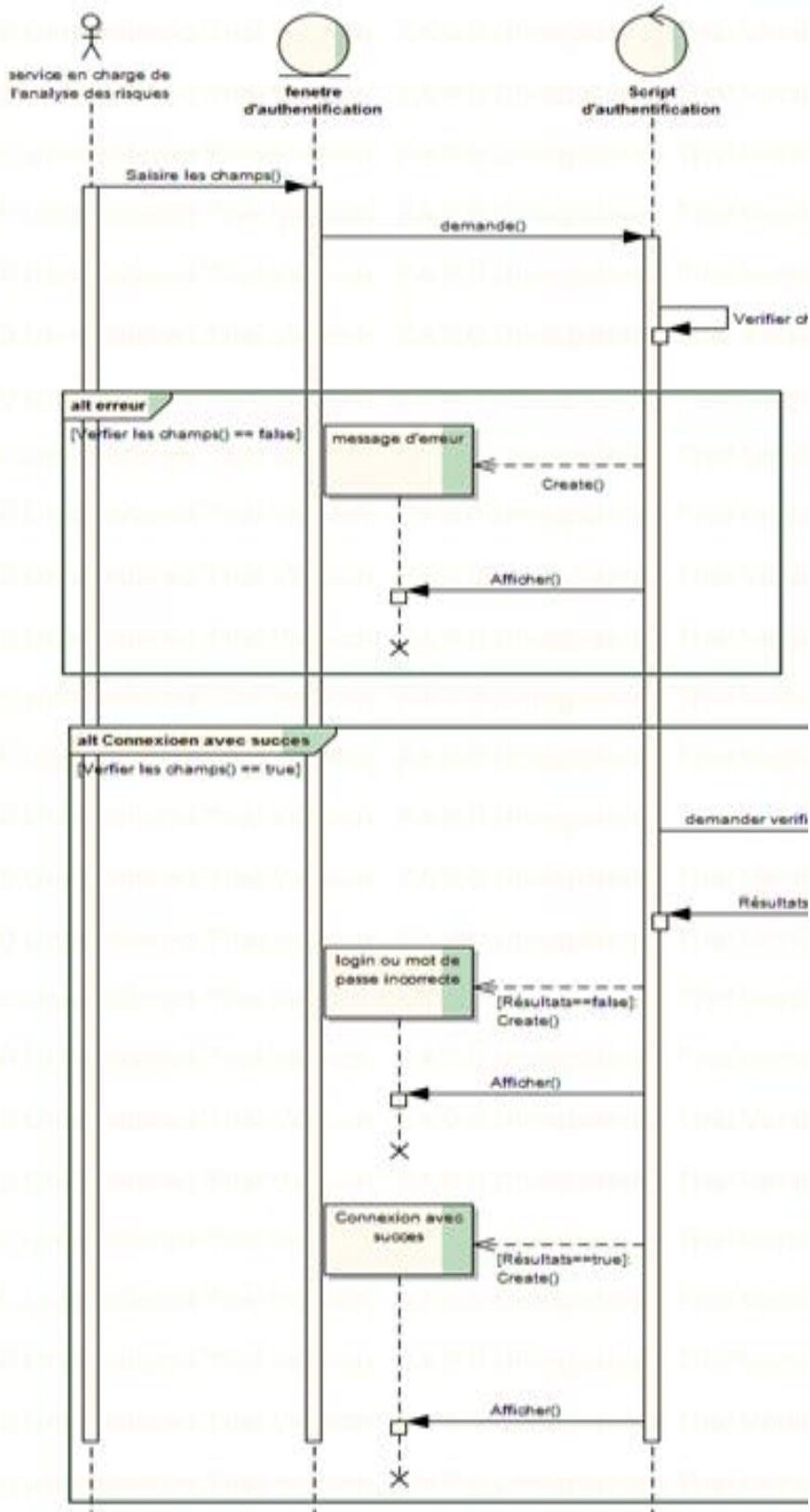


Figure 9 : Diagramme de séquence d'authentification



Diagramme de séquence d'ajout d'un risque :

- Acteur principal : utilisateur (service en charge de l'analyse des risques).
- Intention de l'acteur : l'utilisateur peut ajouter, modifier, supprimer, consulter les risques.

Pour le cas d'utilisation d'ajout d'un risque on trouve les scénarios suivants :

- Prés condition : l'utilisateur est obligé de s'authentifier
- Post condition : l'utilisateur valide la saisie et quitte la fenêtre d'ajout.
- Scénario nominal : l'utilisateur saisie les champs et clique sur le bouton valider.
- Scénario alternatif : l'utilisateur n'as pas bien saisie les champs ou bien il a ajouté un risque existant.

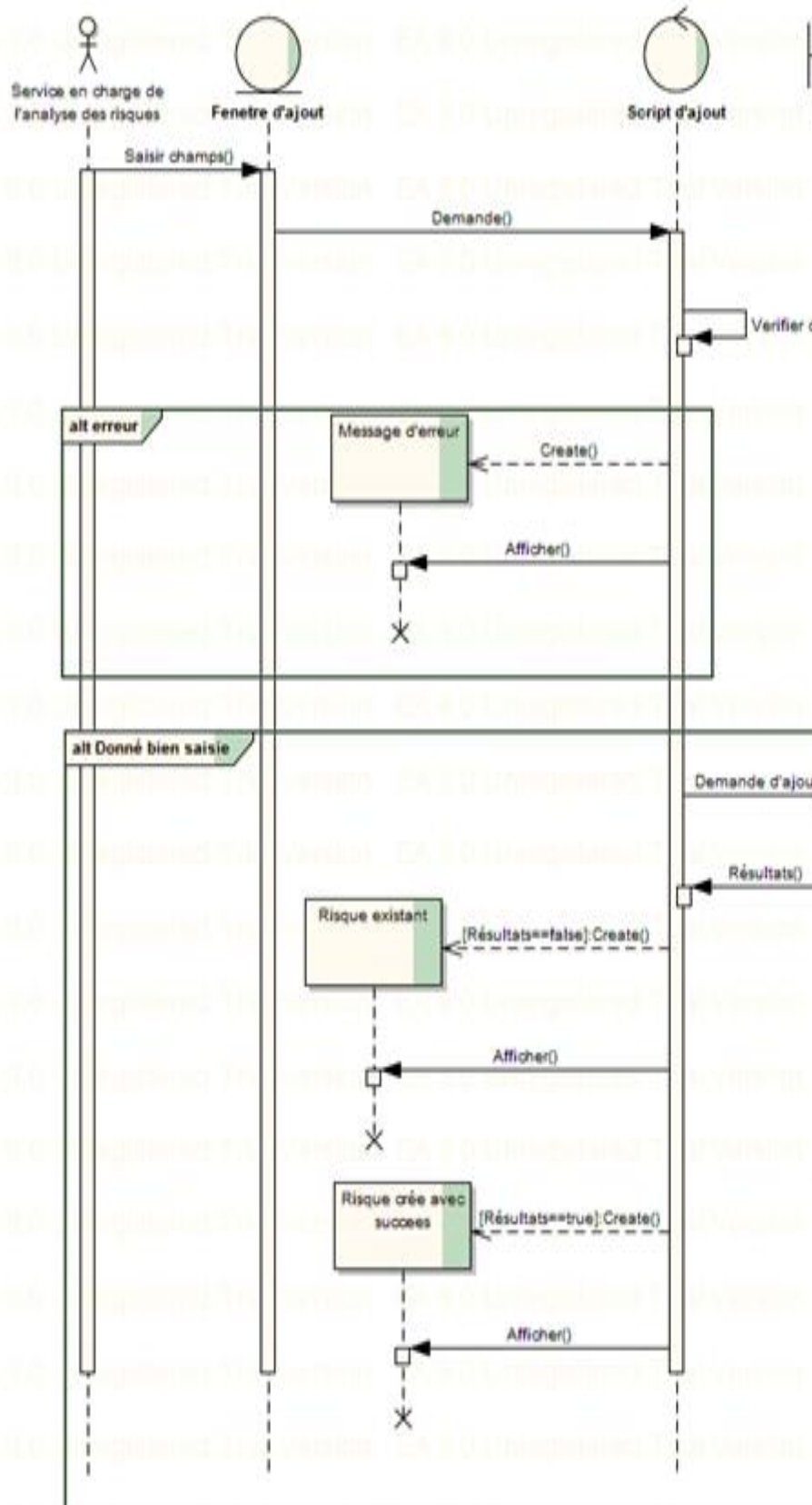


Figure 10 : Diagramme de séquence d'ajout d'un risque



Diagramme d'ajout d'un utilisateur :

- Acteur principal : Administrateur
- Intention de l'acteur : l'administrateur peut ajouter, modifier, supprimer, consulter les utilisateurs.

Pour le cas d'utilisation d'ajout d'un utilisateur, on trouve les scénarios suivants :

- Prés condition : l'administrateur est obligé de s'authentifier
- Post condition : l'administrateur valide la saisie et quitte la fenêtre d'ajout.
- Scénario nominal : l'administrateur saisie les champs et clique sur le bouton valider.
- Scénario alternatif : l'administrateur n'as pas bien saisi les champs ou bien il a ajouté un utilisateur existant.

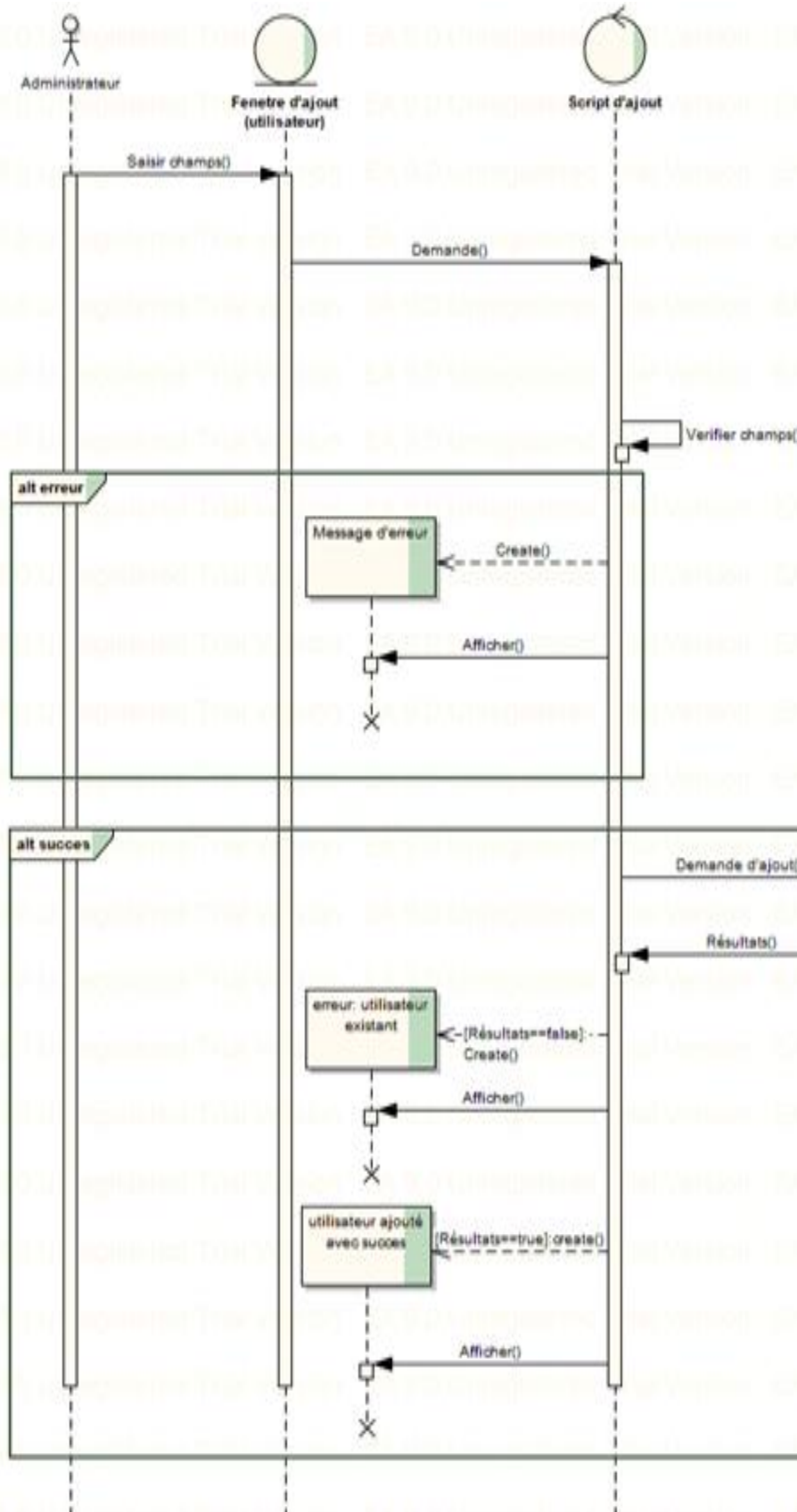


Figure 11 : Diagramme de séquence d'ajout d'un utilisateur

2.4. Diagramme de classe

Le diagramme de classe est un schéma utilisé en génie logiciel pour présenter les classes et les interfaces des systèmes ainsi que les différentes relations entre celles-ci. Ce diagramme fait partie de la partie statique d'UML car il fait abstraction des aspects temporels et dynamiques. Une classe décrit les responsabilités, le comportement et le type d'un ensemble d'objets. Les éléments de cet ensemble sont les instances de la classe. Une classe est un ensemble de fonctions et de données (attributs) qui sont liées ensemble par un champ sémantique. Les classes sont utilisées dans la programmation orientée objet. Elles permettent de modéliser un programme et ainsi de découper une tâche complexe en plusieurs petits travaux simples. Les classes peuvent être liées entre elles grâce au mécanisme d'héritage qui permet de mettre en évidence des relations de parenté. D'autres relations sont possibles entre des classes, chacune de ces relations est représentée par un arc spécifique dans le diagramme de classes. Elles sont finalement instanciées pour créer des objets.

Et à partir de la description détaillée des cas d'utilisation, nous avons créé le diagramme de classe suivant :

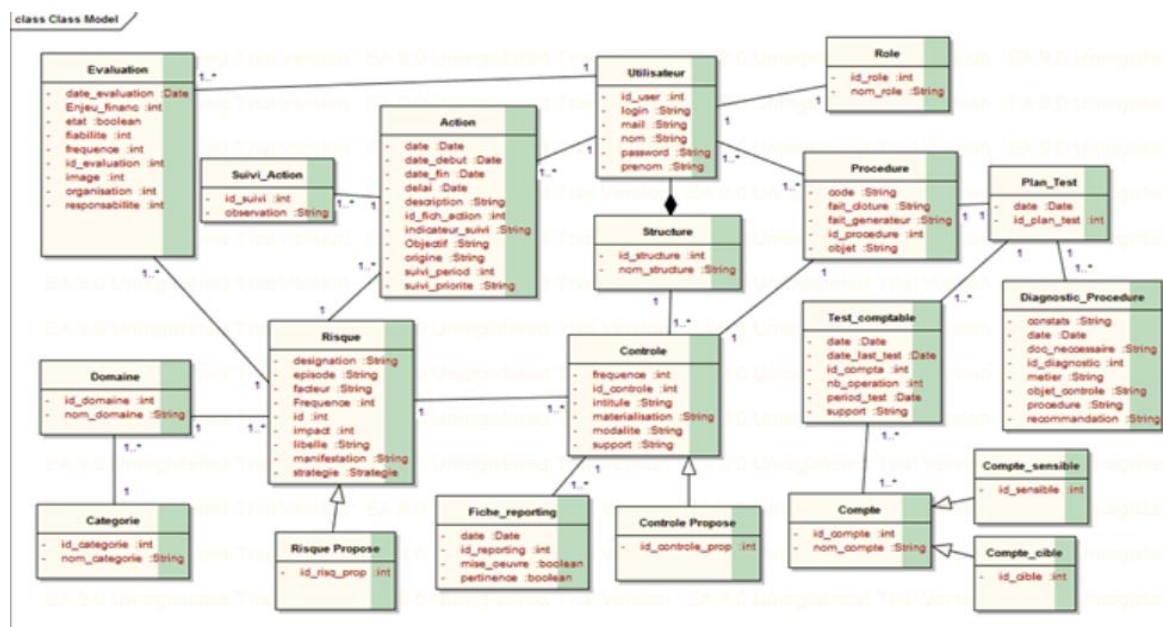


Figure 12: Diagramme de classe



La gestion des risques se base sur la classe « Risque » conçu et mis en œuvre dans le but de traiter les risques auxquels seraient exposées les activités de la « TGR » :

- ✓ Un risque ou plusieurs risques appartient à un domaine
- ✓ Un risque donne naissance à un nouveau risque proposé
- ✓ Un risque subit une ou plusieurs évaluations
- ✓ Un risque est lié à un ou plusieurs contrôles
- ✓ Un risque subit une ou plusieurs actions correctives

Le contrôle interne se base sur la classe « contrôle » mise en œuvre dans le but de réduire les risques susceptibles d'affecter la réalisation des objectifs de la « TGR » :

- ✓ Un ou plusieurs contrôles est affecté à un seul risque
- ✓ Un contrôle est relatif à une procédure
- ✓ Un ou plusieurs contrôles sont créés par une structure
- ✓ Un contrôle donne naissance à un nouveau contrôle

3. ETUDE TECHNIQUE

Conformément à la démarche choisie, cette partie sera consacrée à la branche technique du projet. En fait, le choix de l'architecture technique s'appuie sur les objectifs liés aux besoins techniques de l'application.

3.1. Architecture physique

L'architecture physique assure que les différentes composantes logicielles, déployées à différents endroits du système informatique, collaborent afin de fournir les fonctionnalités attendues.

Nous enchaînons maintenant par la description de l'architecture physique sur laquelle s'exécutent les différents composants logiciels des modules cibles de notre projet, ainsi que la façon par laquelle ils communiquent entre eux.

L'architecture la mieux adaptée à cette application est une architecture client/serveur 3-tiers, vu qu'elle sépare les trois couches logicielles de l'application.

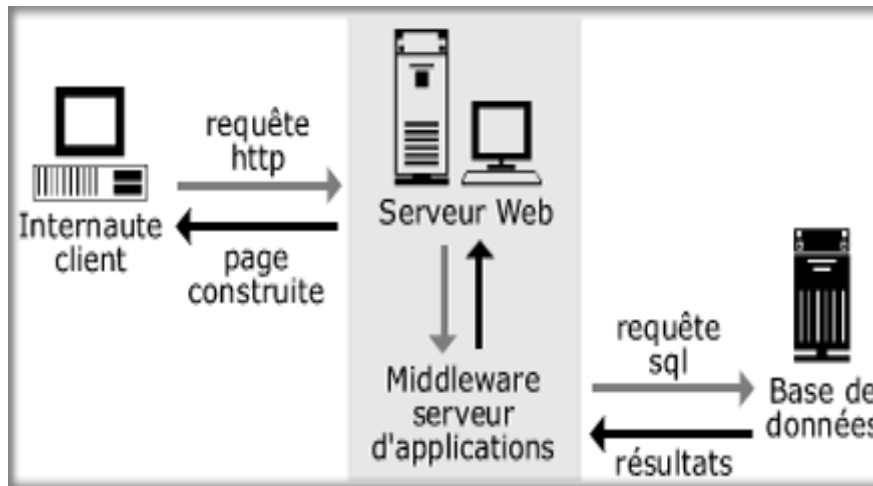


Figure 13 : Architecture physique du projet

La solution à réaliser va intégrer :

- ✓ **Un client** : équipé d'une interface utilisateur (généralement un navigateur web) chargée de la présentation.
- ✓ **Un serveur d'application (appelé également middleware)**: qui fournit la ressource, tout en faisant appel à un autre serveur.
- ✓ **Un serveur de base de données**: fournit au serveur d'application les données requises pour répondre au client.

3.2. Architecture logique

Toujours dans l'optique de séparer les données, les traitements et la présentation, nous avons opté pour l'architecture logique MVC2. Ainsi, l'application se retrouve décomposée en cinq couches essentielles :

- ✓ **View** : couche spécifique aux interfaces de l'application et qui comprend principalement les pages et formes web (.jsp)
- ✓ **Controller** : le contrôleur traite les différentes requêtes d'utilisateurs faisant appel aux divers composants de l'application. Chaque requête atteint d'abord la classe contrôleur frontal. Cette dernière inspecte les demandes entrantes et les achemine à la classe appropriée. Dans

l'architecture que nous avons adopté (MVC2), ce composant figure sous la notation [Dispatcher Servlet] mais encapsulé par le Framework utilisé.

- ✓ **Model** : représente le retour préparé par le contrôleur pour servir une vue donnée, cette dernière est définie à l'aide des annotations ou dans un fichier de mapping[struts.xml].Généralement des classes actions.
- ✓ **Métier** : implémente les algorithmes « métier » de l'application. Cette couche est indépendante de toute forme d'interface directe avec l'utilisateur. Elle est chargée d'exécuter les fonctionnalités métiers de l'application. Constituée essentiellement des entités images extraites à partir de la base de données.
- ✓ **DAO** : la couche d'accès aux données, s'occupe de l'implémentation des opérations CRUD (Create Retrieve Update Delete) sur les objets persistants.

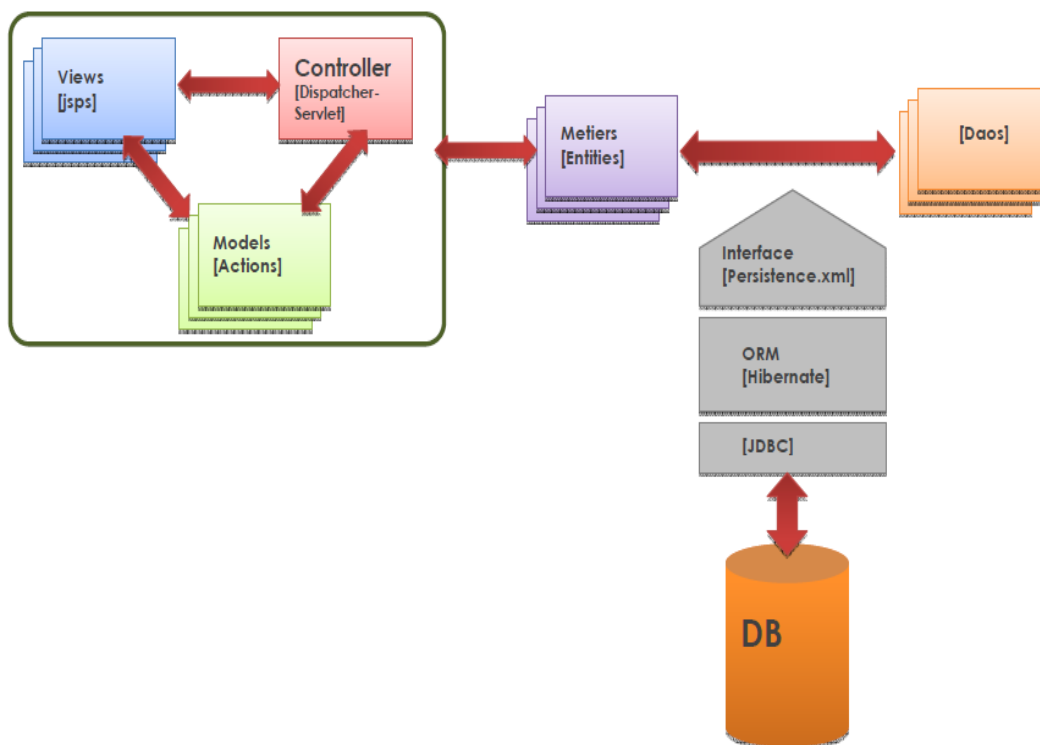


Figure 14 :L'architecture en couche de notre application



IV. CHAPITRE 3 : MISE EN ŒUVRE DU PROJET

Ce chapitre est essentiellement consacré à la présentation des différents outils utilisés lors de la mise en œuvre de cette application et des principales interfaces du système sous forme de capture d'écrans. Chaque écran est précédé d'un commentaire pour décrire son fonctionnement.

1. REALISATION

Nous avons opté pour l'utilisation des technologies suivantes notamment pour bénéficier de l'expérience du cadre professionnel présent à la Trésorerie Générale du Royaume.

1.1. Environnement de développement Eclipse

Eclipse est un IDE, *Integrated Development Environment* (EDI environnement de développement intégré en français), c'est-à-dire un logiciel qui simplifie la programmation en proposant un certain nombre de raccourcis et d'aide à la programmation. Développé par IBM, ce logiciel est gratuit et disponible pour la plupart des systèmes d'exploitation.

Son intérêt se situe dans la facilitation de la mise en place de différents Framework.

1.2. Serveur web et d'application : Apache Tomcat

Nous nous sommes servis de Tomcat comme serveur d'application de notre projet. Comme il inclut un serveur http interne, il est aussi considéré comme serveur http. Tomcat est souvent utilisé en association avec un autre serveur web, en général Apache.

1.3. SGBD MySQL :

MySQL est un SGBD qui fait partie des logiciels de gestion des bases de données les plus utilisés dans le monde, autant par le grand public (application web principalement) que par les professionnels.

Nous avons choisi MySQL comme serveur de bases de données relationnelles, parce qu'il est :

- OpenSource.
- Multithread et multiutilisateurs.



- Très courant surtout pour les applications Web.
- Facilement intégrable dans un environnement Apache Tomcat.
- Pas gourmand en ressource mémoire et en espace disque.

1.4. Architecture J2EE :

(Java 2 Enterprise Edition) est une norme proposée par la firme Sun, visant à définir un standard de développement d'applications d'entreprises multi niveaux, basées sur des composants. J2EE s'appuie entièrement sur Java.

L'architecture J2EE repose sur des composants distincts, interchangeables et distribués, ce qui permet :

- Une simplicité d'extension de l'architecture.
- Une haute disponibilité assurant une bonne qualité de service.
- Une maintenabilité des applications

1.5. Framework de persistance : Hibernate

Nous avons choisi Hibernate vu qu'il est un Framework open source qui nous permettra de réaliser, d'une manière automatique et en utilisant le serveur d'application Tomcat, le mapping entre les objets et les données stockées dans une base de données relationnelle. Son utilisation optimisera le temps réservé par le développement. C'est aussi un Framework qui permet facilement la migration vers une autre BD.

1.6. Spring MVC :

Le Framework Spring est un conteneur dit « léger », c'est-à-dire une infrastructure similaire à un serveur d'application J2EE. Il prend donc en charge la création d'objets et la mise en relation d'objets par l'intermédiaire d'un fichier de configuration qui décrit les objets à fabriquer et les relations de dépendance entre ces objets.

L'avantage par rapport aux serveurs d'application est qu'avec SPRING, les classes n'ont pas besoin d'implémenter une quelconque interface pour être prises en charge par le Framework.

1.7. Framework de présentation : JPA

La Java Persistence API (abrégée en JPA), est une interface de programmation Java permettant aux développeurs d'organiser des données relationnelles dans



des applications utilisant la plateforme Java. La Java Persistence API est à l'origine issue du travail du groupe d'experts JSR 220.

Dans ce contexte, la persistance recouvre 3 zones :

- l'API elle-même, définie dans le paquetage *javax.persistence*
- le langage Java Persistence Query (JPQL)
- l'objet/les métadonnées relationnelles.

La Java Persistence API repose essentiellement sur l'utilisation des annotations introduites dans Java 5. Elles permettent de définir très facilement, et précisément des objets métier, qui pourront servir d'interface entre la base de données et l'application.

1.8. WampServer :

WampServer est une plate-forme de développement Web sous Windows. Elle permet de développer des applications Web dynamiques à l'aide du serveur Apache2, du langage de scripts PHP et d'une base de données MySQL. Elle possède également PHPMyAdmin pour gérer plus facilement les bases de données.

Contrairement aux autres solutions, WampServer permet de reproduire fidèlement le serveur de production. Une fois la base installée, on a la possibilité d'ajouter autant de versions d'Apache, MySQL et PHP que nous le souhaitons.

1.9. Entreprise Architect :

C'est un outil d'analyse et de création UML, couvrant le développement de logiciels du rassemblement d'exigences, en passant par les étapes d'analyse, les modèles de conception et les étapes de test et d'entretien. Cet outil graphique basé sur Windows, peut être utilisé par plusieurs personnes et conçu pour aider à construire des logiciels faciles à mettre à jour. Il comprend un outil de production de documentation souple et de haute qualité.

1.10. Spring Security :

Spring Security propose un modèle de sécurité éprouvé, stable, performant, et répond à l'ensemble des attentes : sécurisation des URL, des méthodes et des

instances d'objets, fourniture de nombreux filtres (par exemple, permet l'authentification par formulaire, l'authentification automatique par cookie, etc...).

Pour des besoins de sécurité basiques, Spring Security propose des fonctionnalités faciles à mettre en œuvre, grâce notamment à son schéma XML dédié. Pour des besoins plus avancés, il est nécessaire d'appréhender des mécanismes plus avancés du framework.

2. PRESENTATION DE L'APPLICATION

Cette partie est essentiellement consacrée à la présentation des principales interfaces du système sous forme de capture d'écrans.

2.1. Formulaire d'authentification

Par mesure de sécurité, l'accès à l'application est protégé par un formulaire d'authentification.

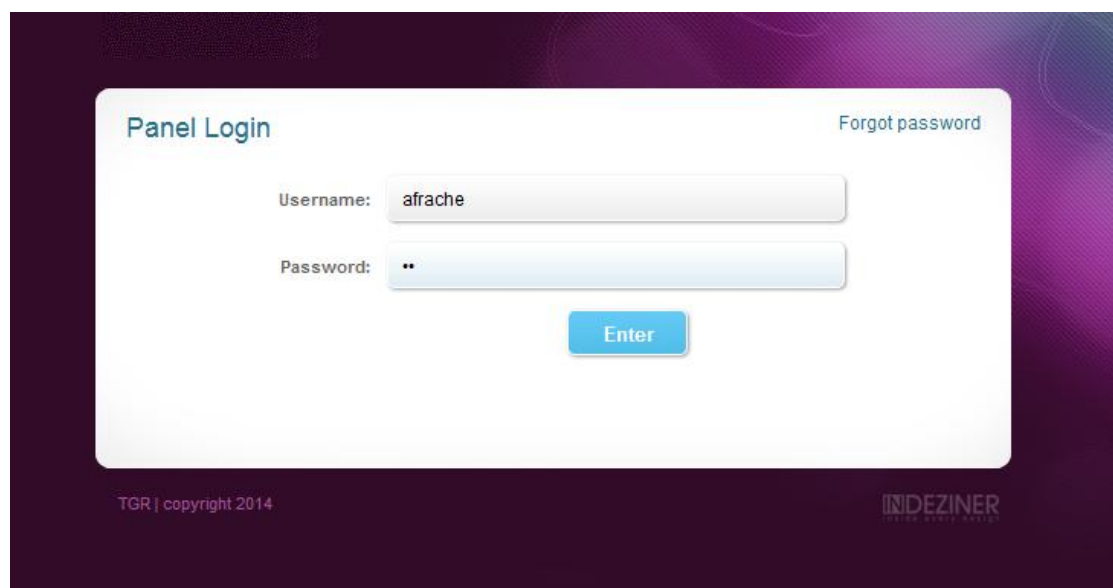


Figure 15 : Page d'authentification

Pour se connecter, l'utilisateur doit saisir son login et son mot de passe. Au cas où les données sont correctes, le système affiche la page d'accueil de chaque utilisateur. Dans le cas contraire, le système affiche un message d'erreur.

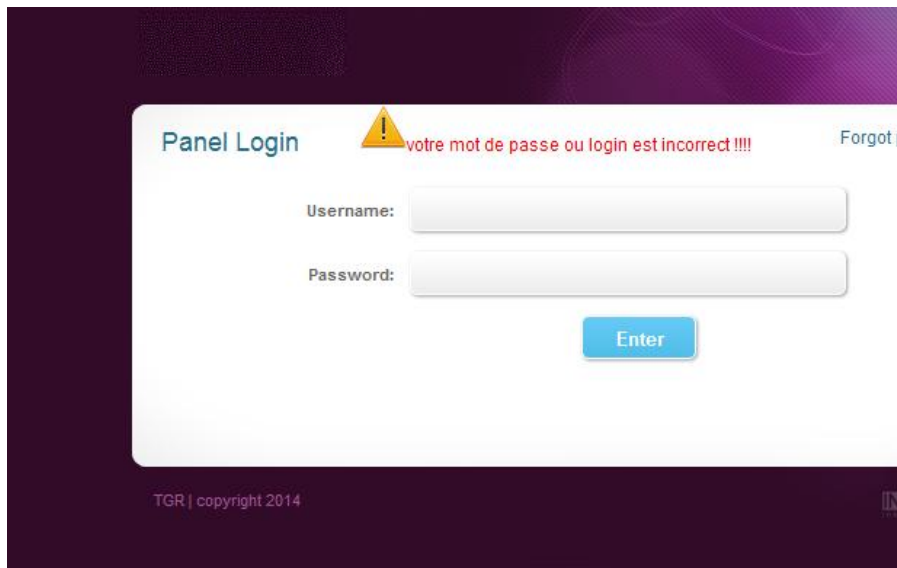


Figure 16 : Echec d'authentification

La page d'accueil du service chargé de l'analyse des risques.

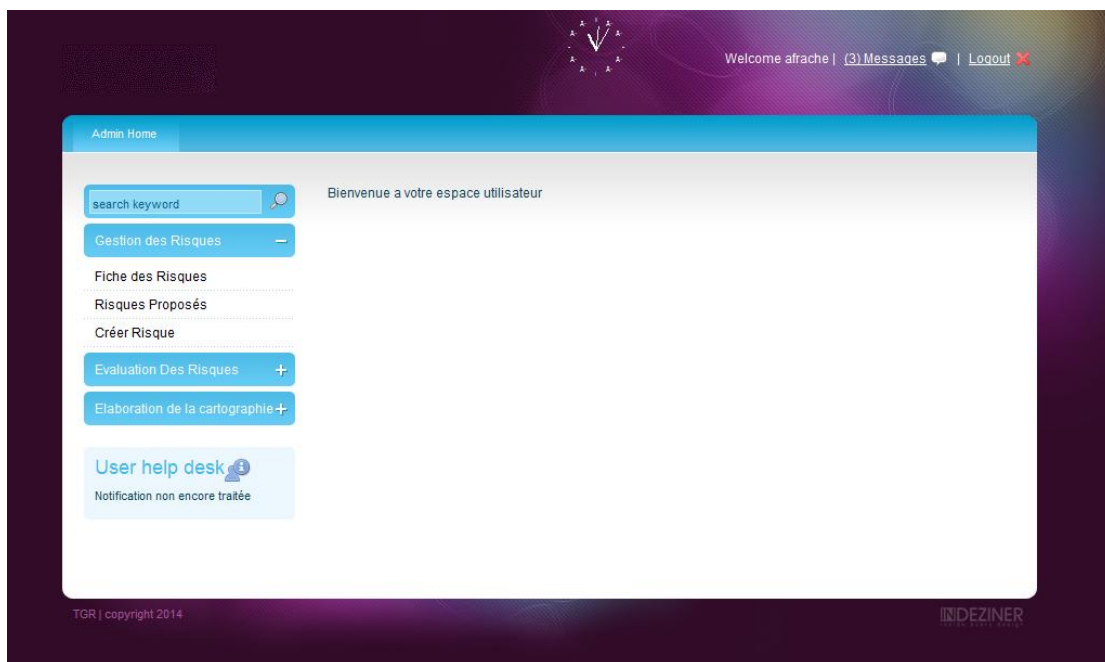


Figure 17 : Service en charge de l'analyse des risques

La page d'accueil du service du contrôle interne.

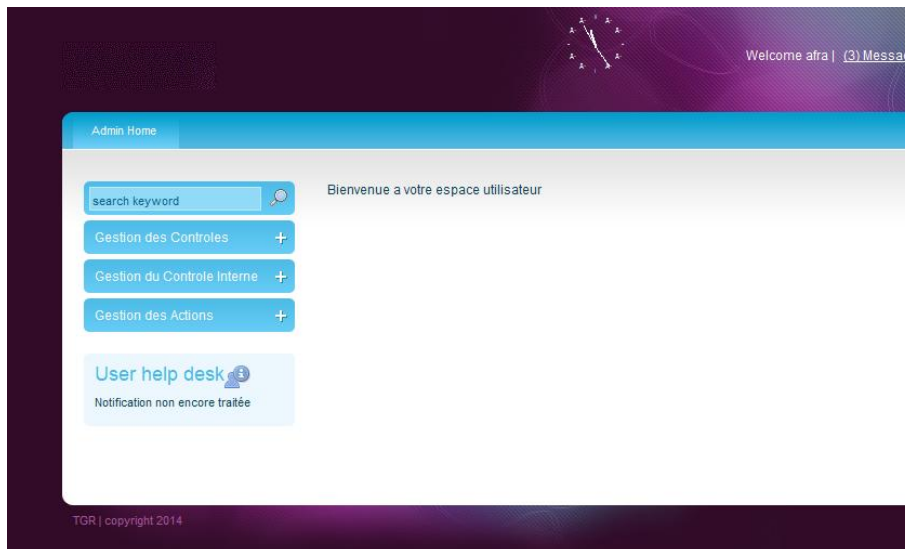


Figure 18 : Service du contrôle interne

La page d'accueil du responsable du contrôle.

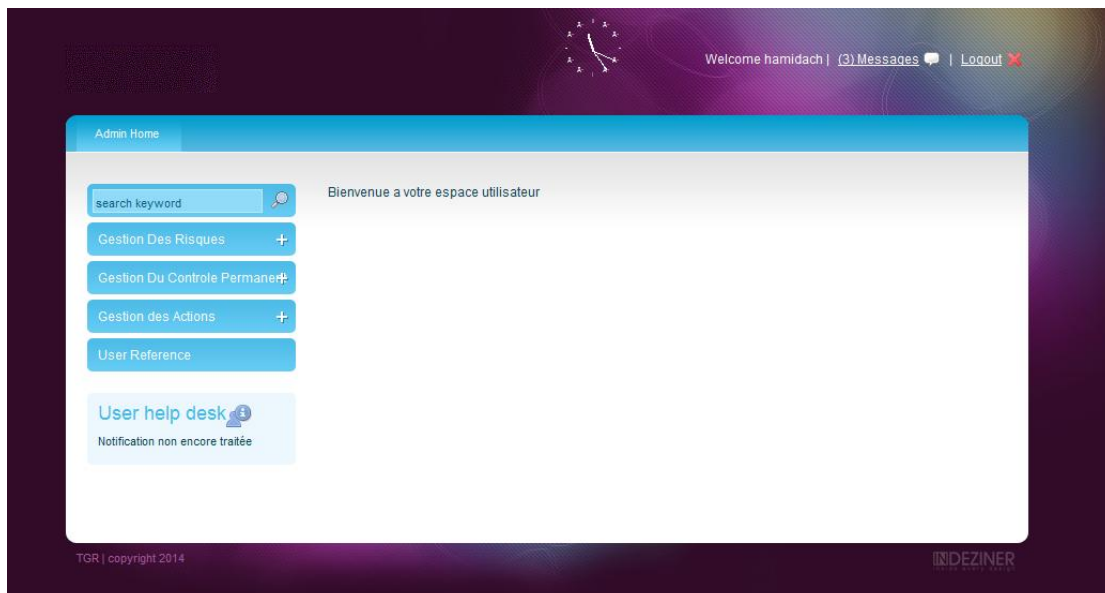


Figure 19 : Responsable du contrôle

2.2. Interface d'ajout d'un risque

Cette interface contient les champs à remplir pour créer un risque.

Admin Home

search keyword

Gestion des Risques

Fiche des Risques

Risques Proposés

Créer Risque

Evaluation Des Risques +

Elaboration de la cartographie +

User help desk ⓘ
Notification non encore traitée

Libelle :

designation:

Domaine: Recouvrement des créances publiques ▼

Manifestation :

Facteur:

Episode du risque:

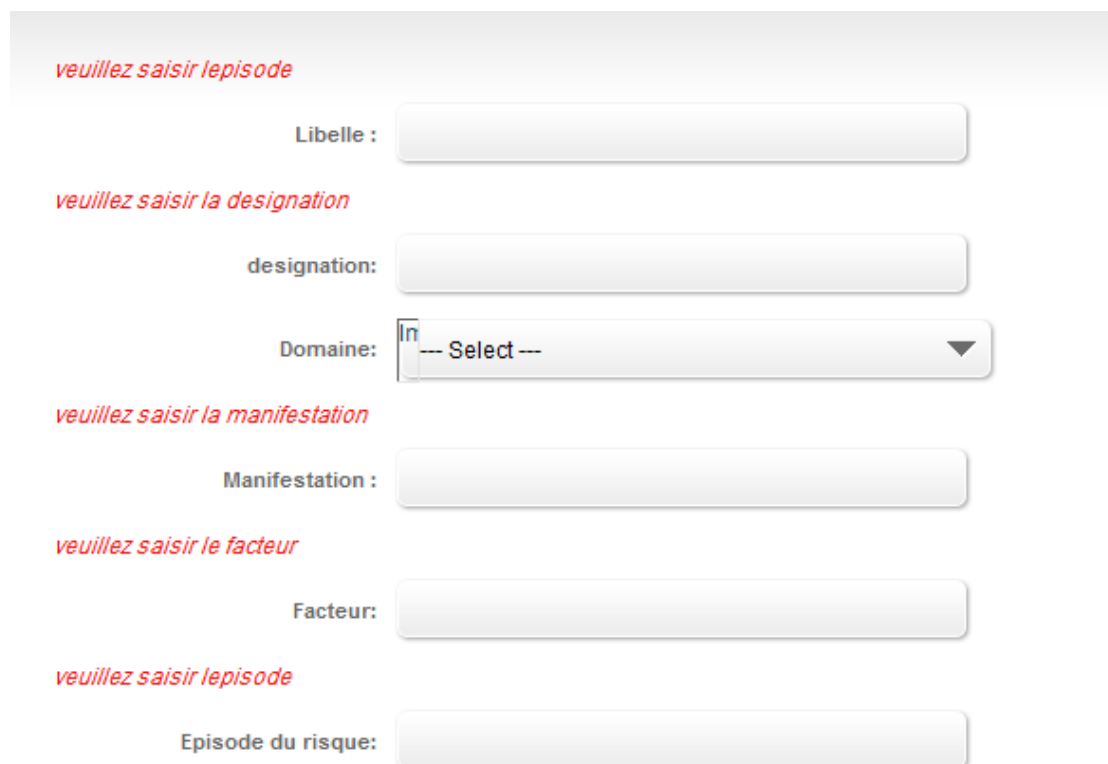
Date:

Frequence:

Impact:

Figure 20 : Créer un risque

Cette interface représente quelques erreurs que l'utilisateur peut commettre lors du remplissage du formulaire précédent.



veillez saisir lepisode

Libelle :

veillez saisir la designation

designation:

Domaine: --- Select ---

veillez saisir la manifestation

Manifestation :

veillez saisir le facteur

Facteur:

veillez saisir lepisode

Episode du risque:

Figure 21 : Erreur d'ajout d'un risque

Si le risque est bien ajouté le système affiche le message suivant.

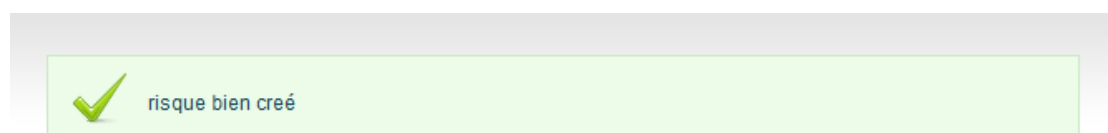


Figure 22 : risque ajouté avec succès

2.3. Fiche des risques

L'interface suivante représente la fiche des risques où l'utilisateur peut modifier et supprimer un risque.

La Liste Des Risques






Libelle	Designation	episode	Date de creation	Edit
lib1	des1	episode1	2015-10-15	
lib2	des2	ep2	2014-02-14	
lib3	des3	epis3	2014-02-15	
lib12	des12	epis12	2015-10-15	

Figure 23 : La fiche des risques

2.4. Suppression d'un risque

Cette interface offre aux utilisateurs la possibilité de supprimer un risque avec un message de confirmation de la suppression.

La Liste Des Risques

Libelle	Designation	episode	Date de creation	Edit	Delete
lib1	des1	episode1	2015-10-15		
lib2	des2	ep2	2014-02-14		
lib3	des3	epis3	2014-02-15		
lib12	des12	epis12	2015-10-15		

Are You Sure ?
Yes Cancel

Figure 24 : Supprimer un risque

La Liste Des Risques

Libelle	Designation	episode	Date de creation	Edit
lib1	des1	episode1	2015-10-15	
lib2	des2	ep2	2014-02-14	
lib3	des3	epis3	2014-02-15	


Figure 25 : Fiche après suppression

2.5. Interface d'ajout d'un contrôle

Cette interface contient les champs à remplir pour créer un contrôle.

intitule :

materialisation :

Risque : 

modalite :

support :

frequence :

Figure 26 : Ajout d'un contrôle

2.6. Interface d'ajout d'une procédure

L'interface suivante contient les champs à remplir pour ajouter une procédure.

Code : code15

objet : obj15

Controle : intitul15

Responsable d Controle : hamidach

Relai Local : afraa>

Fait Generateur : faitg15

Fait de Cloture : faitc15

Ajouter

Figure 27 : Définir une procédure

L'interface suivante affiche la liste des procédures.

La Liste Des Procedures

code	objet	Controle	Edit	Delete
code1	obj1	intitl1		
code2	obj2	intitul2		
code3	obj3	intitul3		
code4	obj4	intitul4		

Figure 28 : La fiche des procédures.

2.7. L'envoi de demande d'évaluation

Le service chargé de l'analyse des risques, envoie à chaque fois, une demande d'évaluation au responsable du risque pour l'évaluer à travers l'interface suivante.

La Liste Des Risques

Libelle	Designation	episode	Date de creation	Demande d'evaluation	
<input type="checkbox"/>	lib2	des2	ep2	2014-02-14	Evaluer

Figure 29 : Envoyer une demande d'évaluation

S'il n'y a pas de risque à évaluer le système affiche le message suivant.

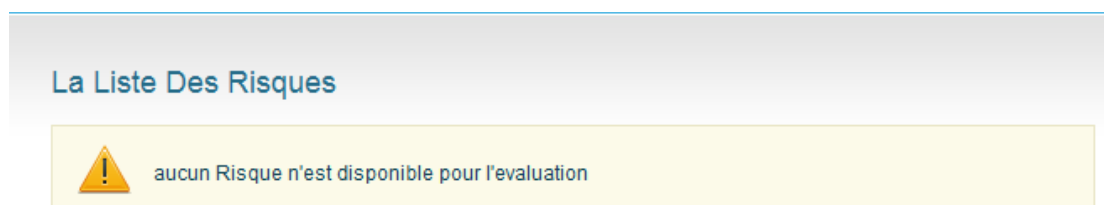


Figure 30 : la liste des risques à évaluer

Les interfaces suivantes représentent l'évaluation des risques par le responsable du contrôle.

Evaluer Risque

Nom du Risque :

Figure 31 : Evaluer risque

enjeu_financ : 3

fiabilite : 2

image : 1

organisation : 1

responsabilite : 2

frequence : 5

date_evaluation : 15/10/2015

[Valider](#)

Figure 32 : Valider l'évaluation

2.8. Traitement d'évaluation

Pour le traitement d'évaluation, le service chargé de l'analyse des risques traite les risques évalué par le responsable. L'interface suivante représente ce traitement.

Libelle	Designation	Nombre des evaluations	Action
lib2	des2	1	Traiter
lib1	des1	2	Traiter
lib3	des3	1	Traiter

Figure 33 : Les risques à traiter

Dans le cas où les risques sont en cours de traitement, le système affiche le message suivant.

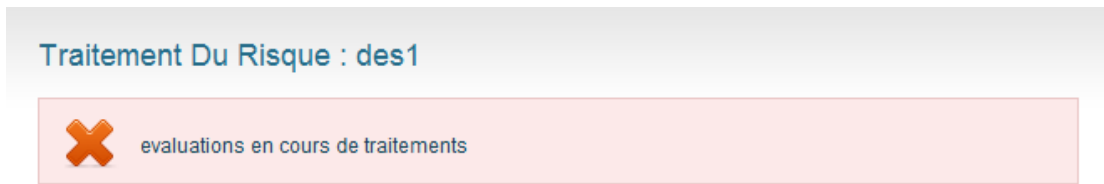


Figure 34 : Evaluation en cours de traitement

Les interfaces suivantes représentent le traitement d'un risque. Ce traitement peut prendre la forme soit d'une annulation ou d'une validation de l'évaluation, soit la forme d'un affichage détaillant un risque.

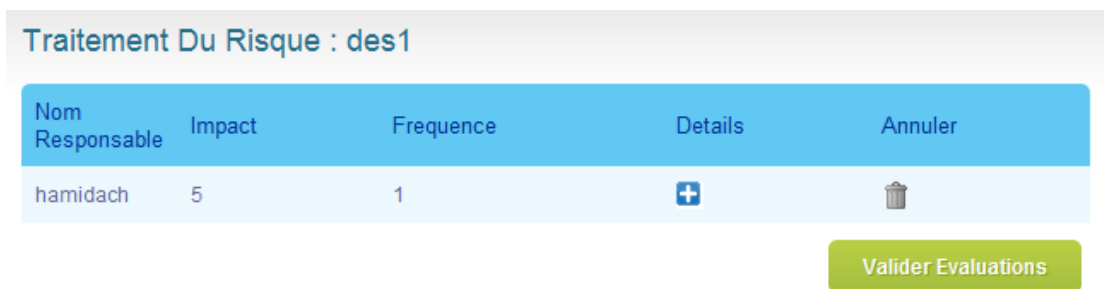


Figure 35 : Traiter évaluation

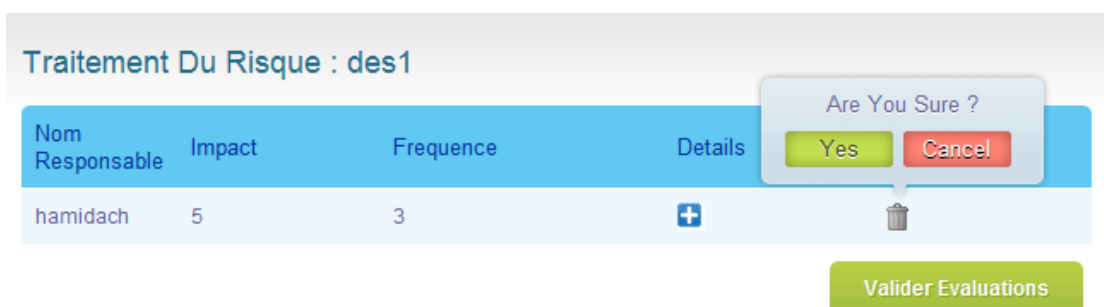


Figure 36 : Annuler une évaluation

Traitement Du Risque : des1

Nom du Responsable	Frequence	Enjeu Financier	Image	Fiabilite	Organisation
hamidach	5	3	3	2	1

[Retour](#)

Figure 37 : Détail d'une évaluation

2.9. Suivi d'action

Pour le suivi d'action, le service du contrôle interne envoie l'action créée au responsable chargé de faire le suivi.

Les interfaces ci-après illustrent le suivi d'action :

Suivi Des Actions

Nom d'action :

date d'observation :

Realisation :

observation :

Figure 38 : Effectuer un suivi

description	Indicateur de suivi	Date Observation	Realisation	Observation
desc1	ind1	2014-05-14	dada	obs1

Figure 39 : Affichage d'un suivi

2.10. Création de la fiche de reporting

Le responsable du contrôle prépare une fiche de reporting pour mesurer la pertinence d'un contrôle et formule une observation sur ce dernier.

Ces interfaces représentent l'ajout d'une fiche de reporting et l'affichage de celle-ci.

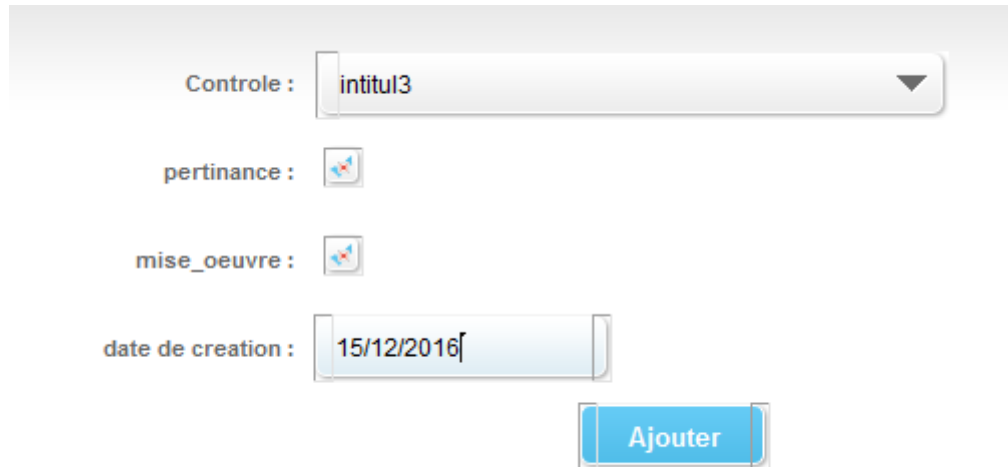


Figure 40 : Création de la fiche de reporting

Domaine	Risque	Controle	Pertinance	Mise-en-oeuvre
Activité bancaire	des1	intit1	true	true

Figure 41 : Affichage de la fiche de reporting



V. CONCLUSION ET PERSPECTIVES

A l'issue de ce projet de fin d'étude, nous tenons de prime abord à exprimer notre entière satisfaction par rapport au déroulement de ce stage pratique à la Trésorerie Générale du Royaume.

En effet, cette opportunité nous a permis de se familiariser d'avantage avec les bases des frameworks les plus couramment utilisés tels que Spring et Hibernate.

En outre, et bien que la durée du stage fût relativement courte, l'objectif que nous sommes fixés a été globalement escompté. Le stage était ainsi une occasion de centrer nos travaux sur le développement J2EE, qui domine aujourd'hui l'environnement et le domaine de l'informatique.

Ce travail reste toutefois perfectible. En effet, et eu égard à la complexité des APIs, l'approche que nous avons adopté pourrait être étendue dans le cadre de travaux futurs.

Au-delà des aspects purement techniques, le stage effectué à la Trésorerie Générale du Royaume nous a offert d'une part, l'occasion d'enrichir nos connaissances théoriques et d'autre part, de les mettre à l'épreuve dans un environnement professionnel de très grande qualité.



BIBLIOGRAPHIE

1. PARTIE DOCUMENTATION SUR LE CONTROLE INTERNE ET LA GESTION DES RISQUES

- Cadre de référence CI - TGR - 20 02 2013 .pdf
- Guide Methodologique - Contrôles périodiques - 10 10 2012 .pdf
- TGR Procédure - contrôles permanents 10-10-2012.pdf
- TGR Procédure–contrôlespériodiques 10 10 2012.pdf
- Guide Méthodologique - Elaboration et suivi des PAF - 10 10 2012.pdf

2. PARTIE DU DEVELOPPEMENT

- SpringByExample auteur David Winterfeldt
- COURS-HIBERNATE-4p auteur Claude Duvallet
- Spring par la pratique 2n édition [www.worldmediafiles.com]
- hibernate_reference auteur Gavin King



WEBOGRAPHIE

- <http://general.developpez.com/edi/>
 - <http://simonhazout.wordpress.com/>
 - <http://fr.wikipedia.org/wiki>
 - <http://www.jmdoudoux.fr/java/dejae/>
 - <http://docs.spring.io/spring/docs/2.5.5/reference/>
 - <http://www.dineshonjava.com/>
 - <https://www.tgr.gov.ma/wps/portal>
 - <http://docs.jboss.org/hibernate/stable/core.old/reference/fr/html>
- ∟