

**UNIVERSITE DE FIANARANTSOA**

**ECOLE NATIONALE D’INFORMATIQUE**

**MEMOIRE DE FIN D’ETUDES POUR L’OBTENTION DU DIPLOME MASTER PROFESSIONNEL EN INFORMATIQUE**

Parcours : Informatique Générale

**Intitulé :**

**CONCEPTION ET REALISATION D’UNE PLATEFORME WEB DE GESTION DE PARC IMMOBILIER**

**(Module Financement)**

Présenté le , par RABARISON HELIODORE Lewis Campbell

Membre du jury :

* Rapporteur
* Examinateur
* Rapporteur : - Monsieur **RAKOTONOMENJANAHARY** Niaina, Encadreur professionnel

Année universitaire : 2019 - 2020

CURRICULUM VITAE



RABARISON HELIODORE Lewis Campbell

Lot 3604 II 002 Antsororokavo Fianarantsoa

Tel: 034 71 194 83

Email : rabarison.lewis@yahoo.com

24 ans, Célibataire

**FORMATIONS ACADEMIQUES ET DIPLOMATIQUES**

**2019-2020** - Cinquièmes années de formation en master professionnelle à l’Ecole Nationale d’Informatique Fianarantsoa

**Parcours** : Informatique Générale

**2018-2019** - Quatrièmes années de formation en master professionnelle à l’Ecole Nationale d’Informatique Fianarantsoa

**Parcours :** Informatique Générale

**2017-2018** - Troisièmes années de formation en licence professionnelle à l’Ecole Nationale d’Informatique Fianarantsoa

**Parcours :** Informatique Générale

**STAGES**

**2017-2018** - Stagiaire au sein du SSII Livenexx Antananarivo.

* + Thème : Conception et réalisation d’une plateforme de gestion de parc immobilier.

**2017-2018** - Stagiaire au sein du Service du Renseignement Financier SAMIFIN Antananarivo.

* + Thème : Conception et réalisation d’une application web pour la gestion de stock des articles.

**2016-2017** - Stagiaire au sein du Ministère de l’Education Nationale Antananarivo

* + Thème : Conception et réalisation d’un Bot Messenger pour aider les fonctionnaires sur la préparation des dossiers ressources humaines.

**EXPERIENCES PROFESSIONNELLES**

**2019 – 2020** - Développement d’une application Android de gestion de réservation d’avion avec Flutter.

**2018 – 2019** - Conception et réalisation d’une application web de gestion de matériels et de comptabilité de matière avec Symfony.

**2017 – 2018** - Développement d’application web de gestion des heures complémentaires avec Java.

**CONNAISSANCES EN INFORMATIQUE**

**Systèmes d’exploitation** : Windows, Linux

**Système de Gestion de Base de Données** : MySQL, PostgreSQL

**Langage de programmation** : C/C++, Java.

**Technologie web:** HTML 5, CSS 3, Bootstrap, Materialize, JavaScript, JQuery, React.js, Angular, Express.js, PHP, CodeIgniter, Symfony.

**Conceptions** : MERISE ,2TUP, Notation UML.

**Outils de developpemt :** PhpStorm,Visual Studio Code,Sublime Text,Netbeans

**CONNAISSANCES LINGUISTIQUES**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Langues | Comprendre à l’audition | | | | | Lire, écrire et rédaction | | | | | Parler et communiquer  Oralement | | | | |
| Niveau de connaissance | TB | B | AB | P |  | TB | B | AB | P | NS | TB | B | AB | P | NS |
| Français |  | X |  |  |  |  | X |  |  |  |  | X |  |  |  |
| Anglais |  | X |  |  |  |  | X |  |  |  |  | X |  |  |  |

**TB** : Très Bien, **B** : Bien, **AB** : Assez Bien, **P** : Passable

**DIVERS**

**Sports pratiques** : Football

**Loisirs** : Surfer sur Internet, regarder des documentaires

SOMMAIRE GENERALE

[CURRICULUM VITAE I](#_Toc80539682)

[SOMMAIRE GENERALE III](#_Toc80539683)

[REMERCIEMENTS IV](#_Toc80539684)

[LISTE DES FIGURES VI](#_Toc80539685)

[LISTE DES TABLEAUX VIII](#_Toc80539686)

[**LISTE DES ABREVIATIONS** IX](#_Toc80539687)

[INTRODUCTION GENERALE 1](#_Toc80539688)

[Partie I : PRESENTATIONS GENERALE 2](#_Toc80539689)

[**Chapitre 1 : PRESENTATION DE L’ECOLE NATIONALE D’INFORMATIQUE** 3](#_Toc80539690)

[Chapitre 2 : Présentation de LIVDHASBJF 17](#_Toc80539691)

[Chapitre 3 : Description du projet 21](#_Toc80539692)

[Partie II : ANALYSE ET CONCEPTION 23](#_Toc80539693)

[Chapitre 4. Analyse préalable 24](#_Toc80539694)

[Chapitre 5 : Analyse conceptuelle 38](#_Toc80539695)

[Chapitre 6 : Analyse conceptuelle 43](#_Toc80539696)

[6.1. Diagramme de classe de conception 43](#_Toc80539697)

[Chapitre 7 : Conception détaillée 65](#_Toc80539698)

[Chapitre 8 : Mise en place de l’environnement de développement 83](#_Toc80539699)

[CONCLUSION 100](#_Toc80539700)

[BIBLIOGRAPHIE X](#_Toc80539701)

[WEBOGRAPHIE X](#_Toc80539702)

[TABLE DE MATIERE XI](#_Toc80539703)

REMERCIEMENTS

Tout d'abord, je remercie Dieu de m'avoir donné la santé, la force à la réalisation de ce travail.

Néanmoins, je tiens à témoigner mes profondes gratitudes à plusieurs personnes sans qui ce présent travail ne s’est pas réalisé.

* Monsieur **RAFAMANTANANTSOA** Fontaine, Professeur à l’université de Fianarantsoa qui m’a accepté et de m’avoir accueilli en tant qu’étudiant.
* Monsieur **RAMAMONJISOA** Andriantiana Bertin Olivier, Professeur titulaire à l’Université de Fianarantsoa, Directeur de l’Ecole Nationale de l’Informatique, qui m’a donné l’opportunité d’approfondir mes connaissances et de développer mes savoir-faire grâce à ce stage.
* Monsieur **HASSIM** Richad, Directeur général de la société LIVENEXX, de m’avoir donnée l’opportunité de faire un stage au sein de la société.
* Monsieur **RAKOTOZOKINY** Njara, Directeur Technique du LIVENEXX de m’avoir accepté en tant que stagiaire au sein de la société.
* Monsieur **RAKOTONOMENJANAHARY** Niaina, Chef de projet Backend de m’avoir encadré tout au long de mon stage au sein de la société.
* Monsieur **ANDRIANAMBININA** Zo, Chef de projet Backend de m’avoir donné de formation au sein de la société.
* Tous les Enseignants qui nous ont transmis leurs connaissances.
* Toute la famille et à tous les amis qui m’a soutenu moralement et financièrement durant la période de stage.

LISTE DES FIGURES

[Figure 1: Organigramme actuel de l’ENI 14](#_Toc79396221)

[Figure 2: Organigramme de Livenexx 25](#_Toc79396222)

[Figure 3 : Processus de développement de la société LIVENEXX 26](#_Toc79396223)

[Figure 4: Cycle de développement en Y de 2TUP 35](#_Toc79396224)

[Figure 5: Acteur 46](#_Toc79396225)

[Figure 6 : Cas d'Utilisation 50](#_Toc79396226)

[Figure 7 : Formalisme d'un diagramme de cas d'utilisation 51](#_Toc79396227)

[Figure 8 : Formalisme d'une inclusion 51](#_Toc79396228)

[Figure 9 : Formalisme d'une extension 51](#_Toc79396229)

[Figure 10 : Diagramme de cas d’utilisation 52](#_Toc79396230)

[Figure 11: Diagramme de séquence pour le cas d’utilisation « Gérer les banques » 54](#_Toc79396231)

[Figure 12 : Diagramme de séquence pour le cas d’utilisation « Gérer les types de paiements » 55](#_Toc79396232)

[Figure 13 : Diagramme de séquence pour le cas d’utilisation « Gérer les devises » 56](#_Toc79396233)

[Figure 14: Diagramme de séquence pour le cas d’utilisation « Gérer les contrats » 57](#_Toc79396234)

[Figure 15: Diagramme de séquence pour le cas d’utilisation « Gérer les statuts de biens » 58](#_Toc79396235)

[Figure 16: Diagramme de séquence pour le cas d’utilisation « Gérer les biens » 59](#_Toc79396236)

[Figure 17: Diagramme de séquence pour le cas d’utilisation « Gérer les entités » 60](#_Toc79396237)

[Figure 18 : Diagramme de séquence pour le cas d’utilisation « Gérer les agences » 61](#_Toc79396238)

[Figure 19: Diagramme de séquence pour le cas d’utilisation « Gérer les terrains » 62](#_Toc79396239)

[Figure 20 : Diagramme de cas d’utilisation technique 63](#_Toc79396240)

[Figure 21: Modèle de domaine du système 65](#_Toc79396241)

[Figure 22 : Mécanisme de l’architecture MVC 66](#_Toc79396242)

[Figure 23: Diagramme de séquence de conception « Gérer les banques » 68](#_Toc79396243)

[Figure 24 : Diagramme de séquence de conception « Gérer les types de paiements » 69](#_Toc79396244)

[Figure 25 : Diagramme de séquence de conception « Gérer les devises » 70](#_Toc79396245)

[Figure 26 : Diagramme de séquence de conception « Gérer les contrats » 71](#_Toc79396246)

[Figure 27 : Diagramme de séquence de conception « Gérer les statuts de biens » 72](#_Toc79396247)

[Figure 28 : Diagramme de séquence de conception « Gérer les biens » 73](#_Toc79396248)

[Figure 29 : Diagramme de séquence de conception « Gérer les agences » 74](#_Toc79396249)

[Figure 30 : Diagramme de séquence de conception « Gérer les entités » 75](#_Toc79396250)

[Figure 31 : Diagramme de séquence de conception « Gérer les terrains » 76](#_Toc79396251)

[Figure 32 : Diagramme de classe de conception de « Gérer les banques » 77](#_Toc79396252)

[Figure 33 : Diagramme de classe de conception de « Gérer les entités» 77](#_Toc79396253)

[Figure 34 : Diagramme de classe de conception de « Gérer les terrains » 77](#_Toc79396254)

[Figure 35 : Diagramme de classe de conception de « Gérer les biens » 78](#_Toc79396255)

[Figure 36 : Diagramme de classe de conception de « Gérer les contrats » 78](#_Toc79396256)

[Figure 37 : Diagramme de classe de conception de l’application 79](#_Toc79396257)

[Figure 38 : Diagramme de paquetage de l’application 80](#_Toc79396258)

[Figure 39 : Diagramme de déploiement de l’application 81](#_Toc79396259)

[Figure 40 : Installation de Xampp 83](#_Toc79396260)

[Figure 41 : Installation du PHPStorm 84](#_Toc79396261)

[Figure 42 : Iinstallation du Visual Paradigm 84](#_Toc79396262)

[Figure 43: Configuration du fichier .env 85](#_Toc79396263)

[Figure 44: Commande à exécuter pour créer une base de données 85](#_Toc79396264)

[Figure 45: Illustration de la base de données crée 85](#_Toc79396265)

[Figure 46: Constantes utilisées pour récupérer les entités 86](#_Toc79396266)

[Figure 47: Les routes de la plateforme 86](#_Toc79396267)

[Figure 48 : Les routes du contrat 87](#_Toc79396268)

[Figure 49 : Contrôleur du contrat 88](#_Toc79396269)

[Figure 50 : Vue du contrat 90](#_Toc79396270)

[Figure 51 : Repository du contrat 92](#_Toc79396271)

[Figure 52 : Script du contrat 93](#_Toc79396272)

[Figure 53 : Liste des banques 94](#_Toc79396273)

[Figure 54 : Liste des devises 94](#_Toc79396274)

[Figure 55 : Liste des entités 95](#_Toc79396275)

[Figure 56 : Liste des biens 95](#_Toc79396276)

[Figure 57 : Liste des terrains 96](#_Toc79396277)

[Figure 58 : Liste des agences 96](#_Toc79396278)

[Figure 59 : Liste des contrats 97](#_Toc79396279)

LISTE DES TABLEAUX

[Tableau 1: Organisation du système de formation pédagogique de l’ENI 14](#_Toc79396195)

[Tableau 2: Liste des formations existantes à l'ENI 16](#_Toc79396196)

[Tableau 3: Débouchés professionnels éventuels des diplômés 21](#_Toc79396197)

[Tableau 4: Moyens matériels à la réalisation du projet 27](#_Toc79396198)

[Tableau 5: Inventaire des moyens matériels 30](#_Toc79396199)

[Tableau 6: Comparaison entre deux méthodes de conception 32](#_Toc79396200)

[Tableau 7: Diagramme Structurel 35](#_Toc79396201)

[Tableau 8: Diagramme de comportement 36](#_Toc79396202)

[Tableau 9: Comparatif d'outils de conception 37](#_Toc79396203)

[Tableau 10: Comparatifs des langages de programmation 38](#_Toc79396204)

[Tableau 11: Comparatifs des SGBD 40](#_Toc79396205)

[Tableau 12 : Comparaison de ces Framework PHP 42](#_Toc79396206)

[Tableau 13 : Dictionnaire de données 43](#_Toc79396207)

**LISTE DES ABREVIATIONS**

AUF : Agence Universitaire de la Francophonie

BNGRC : Bureau national de gestion des Risques et des catastrophes

CARI : Colloque Africain sur la Recherche en Informatique

CHU : Centre Hospitalier Universitaire

CITEF : Conférence Internationale des Ecoles de formation d’Ingénieurs et Technicien d’Expression Française

CNH : Commission Nationale d’Habilitation

COFAV : Corridor Forestier de Fandriana jusqu’à Vondrozo

CSS : Cascading StyleSheet

CUR : Centre Universitaire Régional

DEA : Diplôme d’Etudes Approfondies

ENI : Ecole Nationale d'Informatique

ESPA : Ecole Supérieure Polytechnique d’Antananarivo

ESN : Entreprise de Société Numérique

FID : Fond d'Intervention pour le Développement

FPPSM : Forêts, Parcs et Pauvreté dans le Sud de Madagascar

HTML : HyperText Markup Language

HTTP : HyperText Transport Protocol

INPG : Institut National Polytechnique de Grenoble

INSTAT : Institut National de Statistique

IRD : Institut de Recherche pour le Développement

IREMIA : Institut de Recherche en Mathématiques et Informatique Appliquées

JIRAMA : Jiro sy Rano Malagasy

LMD : Licence – Master – Doctorat

MEN : Ministère de l'Education National

MAEP : Ministère de l'Agriculture, de l'Elevage et de Pêche

MESupRES : Ministère de l'Enseignement Supérieur

MIC : Ministère de l'Industrie et de Commerce

MFB : Ministère du Finance et du Budget

MPTDN : Ministère des postes, de télécommunications et du Développement Numérique

MNP : Madagascar National Parks

MVC : Model-View-Controller

OS : Operating System

PHP : HyperText Preprocessor

PRESUP : Programme de renforcement en l’Enseignement Supérieur

SEPT : Société d’Exploitation du Port de Toamasina

SGBD : Système de Gestion de Base de Données

TIC : Technologies de l’information et de la communication

TVA : Taux à Valeur Ajouté

2TUP : 2 Track Unified Process

UPST : Université Paul Sabatier de Toulouse

UML : Unified Modeling Language

USAID : United State Aid

WWF : World Wide Fun

INTRODUCTION GENERALE

Actuellement, le monde connaît une avance technologique considérable dans tous les secteurs et cela grâce à l'informatique qui est une science qui étudie les techniques du traitement automatique de l'information. Elle joue un rôle important dans le développement de l'entreprise et d'autres établissements.

Ainsi, dans le cadre de cette optimisation, l’entreprise AKADIN a mandaté la société LIVENEXX de concevoir et réaliser une plateforme web pour la gestion de parc immobilier. Avec cette plateforme, le responsable n’aura plus à recourir aux archives sur papier pour savoir les informations sur le parc mais il lui suffit de le consulter via la plateforme.

Ce projet permettra d’établir une application optimisée, facilement maintenable et surtout répond aux exigences de l’utilisateur. Pour atteindre ce but, nous avons utilisé une méthode de conception, un langage de modélisation, un outil de conception, un Framework, et enfin un système de gestion de base de données.

Ce présent mémoire se divise en trois grandes parties. Nous débuterons par les présentations de l'ENI et de la société LIVENEXX ainsi que la description du projet. Nous approfondirons ensuite l'analyse et la conception. En dernière partie, nous décrirons la réalisation du projet.

Partie I : PRESENTATIONS GENERALE

**Chapitre 1 : PRESENTATION DE L’ECOLE NATIONALE D’INFORMATIQUE**

### Informations d’ordre général

L’Ecole Nationale d’Informatique, en abrégé ENI, est un établissement d’enseignement supérieur rattaché académiquement et administrativement à l’Université de Fianarantsoa.

Le siège de l’Ecole se trouve à Tanambao- Antaninarenina à Fianarantsoa.

L’adresse pour la prise de contact avec l’Ecole est la suivante :

Ecole Nationale d’Informatique (ENI) Tanambao, Fianarantsoa. Le numéro de sa boîte postale est 1487 avec le code postal 301. Téléphone : 020 75 508 01. Son adresse électronique est la suivante : [***eni@univ-fianar.mg***. Site](mailto:eni@univ-fianar.mg.%20Site) Web : www. [eni@univ-fianar.mg/eni](mailto:eni@univ-fianar.mg/eni)

### Missions et historique

L’ENI se positionne sur l’échiquier socio-éducatif malgache comme étant le plus puissant secteur de diffusion et de vulgarisation des connaissances et des technologies informatiques.

Cette Ecole Supérieure peut être considérée aujourd’hui comme la vitrine et la pépinière des élites informaticiennes du pays.

L’Ecole s’est constituée de façon progressive au sein du Centre Universitaire Régional (CUR) de Fianarantsoa.

De façon formelle, l’ENI était constituée et créée au sein du (CUR) par le décret N° 83-185 du 24 Mai 1983, comme étant le seul établissement Universitaire Professionnalisé au niveau national, destiné à former des techniciens et des Ingénieurs de haut niveau, aptes à répondre aux besoins et exigences d’Informatisation des entreprises, des sociétés et des organes implantés à Madagascar.

L’ENI a pour conséquent pour mission de former des spécialistes informaticiens compétents et opérationnels de différents niveaux notamment :

* En fournissant à des étudiants des connaissances de base en informatique ;
* En leur transmettant le savoir-faire requis, à travers la professionnalisation des formations dispensées et en essayant une meilleure adéquation des formations par rapport aux besoins évolutifs des sociétés et des entreprises.
* En initiant les étudiants aux activités de recherche dans les différents domaines des Technologies de l’information et de la communication (TIC).

L’implantation de cette Ecole Supérieure de technologie de pointe dans un pays en développement et dans une Province (ou Faritany) à tissu économique et industriel faiblement développé ne l’a pourtant pas défavorisée, ni empêchée de former des spécialistes informaticiens de bon niveau, qui sont recherchés par les entreprises, les sociétés et les organismes publics et privés sur le marché de l’emploi.

La filière de formation d’Analystes Programmeurs a été mise en place à l’Ecole en 1983, et a été gelée par la suite en 1996, tandis que la filière de formation d’ingénieurs a été ouverte à l’Ecole en 1986.

Dans le cadre du Programme de renforcement en l’Enseignement Supérieur (PRESUP), la filière de formation des Techniciens Supérieurs en Maintenance des Systèmes des informatiques a été mise en place en 1986 grâce à l’appui matériel et financier de la Mission Française de coopération auprès de l’Ambassade de France à Madagascar.

Une formation pour l’obtention de la certification CCNA et / ou NETWORK +. Appelée « CISCO Networking Academy » a été créée à l’Ecole en 2002-2003 grâce au partenariat avec CISCO SYSTEM et l’Ecole Supérieure Polytechnique d’Antananarivo (ESPA). Cependant, cette formation n’avait pas duré longtemps.

Une formation de troisième cycle a été ouverte à l’Ecole a été ouverte à l’Ecole depuis l’année 2003 – 2004 grâce à la coopération académique et scientifique entre l’Université de Fianarantsoa pour le compte de l’ENI et l’Université Paul Sabatier de Toulouse (UPST).

Cette filière avait pour objectif de former certains étudiants à la recherche dans les différents domaines de l’Informatique, et notamment pour préparer la relève des Enseignants-Chercheurs qui étaient en poste.

Pendant l’année 2007-2008**,** la formation en vue de l’obtention du diplôme de Licence Professionnelle en Informatique a été mise en place à l’ENI avec les deux options suivantes de formation :

* Génie Logiciel et base de Données.
* Administration des Système et réseaux.

La mise en place à l’Ecole de ces deux options de formation devait répondre au besoin de basculement vers le système Licence – Master – Doctorat (LMD).

Mais la filière de formation des Techniciens Supérieurs en Maintenance des Systèmes Informatiques a été gelée en 2009.

En vue de surmonter les difficultés de limitation de l’effectif des étudiants accueillis à l’Ecole, notamment à cause du manque d’infrastructures, un système de « Formation Hybride » a été mise en place à partir de l’année 2010. Il s’agit en effet d’un système de formation semi-présentielle et à distance avec l’utilisation de la visioconférence pour la formation à distance.

Le système de formation hybride a été ainsi créé à Fianarantsoa ainsi qu’Université de Toliara.

### Organigramme institutionnel de l’ENI

Cet organigramme de l’Ecole est inspiré des dispositions du décret N° 83-185 du 23 Mai 1983.

L’ENI est administrée par un conseil d’Ecole, et dirigée par un directeur nommé par un décret adopté en conseil des Ministres.

Le Collège des enseignants regroupant tous les enseignants-chercheurs de l’Ecole est chargé de résoudre les problèmes liés à l’organisation pédagogique des enseignements ainsi que à l’élaboration des emplois du temps.

Le Conseil Scientifique propose les orientations pédagogiques et scientifiques de l’établissement, en tenant compte notamment de l’évolution du marché de travail et de l’adéquation des formations dispensées par rapport aux besoins des entreprises.

Trois départements de formation caractérisent l’organigramme :

* Le département de formation théorique à l’intérieur de l’Ecole ;
* Le département de formation pratique pour la coordination et la supervision des stages en entreprise et des voyages d’études ;
* Le département de formation doctorale pour l’organisation de la formation de 3ème cycle.

La figure 1 présente l’organigramme actuel de l’ENI.

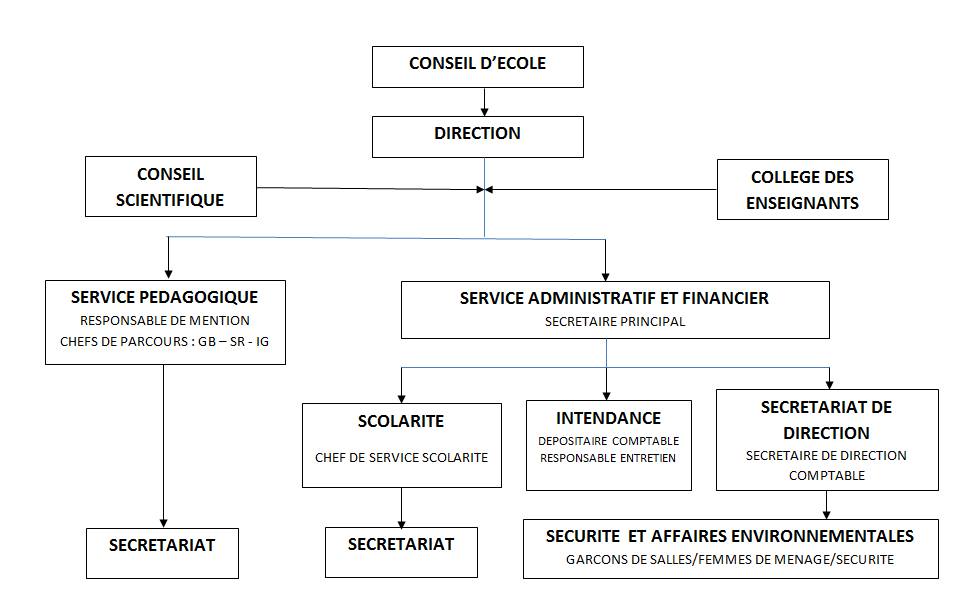


Figure 1: Organigramme actuel de l’ENI

Sur cet organigramme, l’Ecole placée sous la tutelle académique et administrative de l’Université de Fianarantsoa, et dirigée par un Directeur élu par les Enseignants – Chercheurs permanents de l’Etablissement et nommé par un décret pris en Conseil des ministres pour un mandat de 3 ans.

Le Conseil de l’Ecole est l’organe délibérant de l’Ecole.

Le Collège des Enseignants propose et coordonne les programmes d’activités pédagogiques.

Le Conseil scientifique coordonne les programmes de recherche à mettre en œuvre à l’Ecole.

Le Secrétariat principal coordonne les activités des services administratifs (Scolarité, Comptabilité, et Intendance).

Conformément aux textes en vigueur régissant les Etablissements malgaches d’Enseignement Supérieur, qui sont barrés sur le système LMD, les Départements de Formation pédagogique ont été ainsi remplacés par des Mentions et des parcours. Et les chefs des Départements ont été ainsi remplacés par des responsables des mentions et les responsables des parcours.

Un administrateur des Réseaux et Systèmes gère le système d’information de l’Ecole et celui de l’Université.

### Domaines de spécialisation

Les activités de formation et de recherche organisées à l’ENI portent sur les domaines suivants :

* Génie logiciel et Base de Données ;
* Administration des Systèmes et Réseaux ;
* Informatique Générale
* Modélisation informatique et mathématique des Systèmes complexes.

D’une manière plus générale, les programmes des formations sont basés sur l’informatique de gestion et sur l’informatique des Systèmes et Réseaux. Et les modules de formation intègrent aussi bien des éléments d’Informatique fondamentale que des éléments d’Informatique appliquée.

Le tableau 1 décrit l’organisation du système de formation pédagogique de l’Ecole.

Tableau 1: Organisation du système de formation pédagogique de l’ENI

|  |  |
| --- | --- |
| **Formation théorique** | **Formation pratique** |
| * Enseignement théorique * Travaux dirigés * Travaux pratiques | * Etude de cas * Travaux de réalisation * Projets / Projets tutorés * Voyage d’études * Stages |

### Architecture des formations pédagogiques

Le recrutement des étudiants à l’ENI se fait uniquement par voie de concours d’envergure nationale en première année.

Les offres de formation organisées à l’Ecole ont été validées par la Commission Nationale d’Habilitation (CNH) auprès du Ministères de l’Enseignement Supérieur et de la Recherche Scientifique selon les dispositions de l’Arrêté N°31.174/2012-MENS en date du 05 Décembre 2012.

Au sein de l’ENI, il existe une seule mention (INFORMATIQUE) et trois parcours :

* + Génie logiciel et Base de Données ;
  + Administration des Systèmes et Réseaux ;
  + Informatique Générale

L’architecture des études à trois niveaux conforment au système Licence- Master-Doctoral (LMD) permet les comparaisons et les équivalences académiques des diplômes au niveau international.

* L = Licence (Bac + 3) = L1, L2, L3 = 6 semestres S1 à S6
* M = Master (Bac + 5) = M1, M2 = 4 semestres S7 à S10

Le diplôme de licence est obtenu en 3 années des études après Baccalauréat. Et le diplôme de Master est obtenu en 2 ans après obtenu du diplôme de LICENCE.

Le MASTER PROFESSIONNEL est un diplôme destiné à la recherche emploi au terme des études.

Le MASTER RECHERCHE est un diplôme qui remplace l’ancien Diplôme d’Etudes Approfondies (DEA), et qui permet de s’inscrire directement dans une Ecole Doctorale.au terme des études.

* D = Doctorat (Bac +8)

Le Doctorat est un diplôme qu’on peut obtenir en 3 ans après l’obtention du diplôme de MASTER RECHERCHE.

Tableau 2 montre laliste des formations existantes à l’ENI

Tableau 2: Liste des formations existantes à l'ENI

|  |  |  |
| --- | --- | --- |
|  | FORMATION EN | |
|  | LICENCE PROFESSIONNELLE ET HYBRIDE | MASTER |
| Condition d’admission | Par voie de concours Formation Professionnelle : 100 candidats  Formation hybride : 150 candidats |  |
| Condition d’accès | Bac de série C, D ou Technique | Etre titulaire de licence professionnelle |
| Durée de formation | 3 années | 2 années |
| Diplôme à délivrer | Diplôme de Licence Professionnelle en Informatique | Diplôme de Master Professionnel  Diplôme de Master Recherche |

L’accès en première année de MASTER se fait automatiquement pour les étudiants de l’Ecole qui ont obtenu le diplôme de Licence Professionnelle.

Le Master Recherche permet à son titulaire de poursuivre directement des études en doctorat et de s’inscrire directement dans une Ecole Doctorale.

Les Ecoles Doctorales jouissent d’une autonomie de gestion par rapport aux Etablissements de formation universitaire.

Il convient de signaler que par arrêté ministériel N° 21.626/2012 – MESupRES publié le 9 Août 2012 par la Commission National d’habilitation (CNH), l’Ecole Doctorale « Modélisation – Informatique » a été habilitée pour l’Université de Fianarantsoa.

Depuis l’année universitaire 2010-2011, l’ENI s’est mise à organiser des formations hybrides en informatique dans les différentes Régions (Fianarantsoa, Toliara) en raison de l’insuffisance de la capacité d’accueil des infrastructures logistiques. En effet, le système de formation hybride semi - présentielle utilise la visioconférence pour la formation à distance.

Bien qu’il n’existe pas encore au niveau international de reconnaissance écrite et formelle des diplômes délivrés par l’ENI, les étudiants diplômés de l’Ecole sont plutôt bien accueillis dans les instituts universitaires étrangères (CANADA, Suisse, France…)

### Relations de l’ENI avec les entreprises et les organismes

Les stages effectués chaque année par les étudiants mettent l’Ecole en rapport permanent avec plus de 300 entreprises et organismes publics, semi-publics et privés, nationaux et internationaux.

L’Ecole dispose ainsi d’un réseau d’entreprises, de sociétés et d’organismes publics et privés qui sont des partenaires par l’accueil en stage de ses étudiants, et éventuellement pour le recrutement après l’obtention des diplômes par ces derniers.

Les compétences que l’Ecole cherche à développer chez ses étudiants sont l’adaptabilité, le sens de la responsabilité, du travail en équipe, le goût de l’expérimentation et l’innovation.

En effet, la vocation de l’ENI est de former des techniciens supérieurs de niveau LICENCE et des ingénieurs de type généraliste de niveau MASTER avec des qualités scientifiques, techniques et humaines reconnues, capables d’évoluer professionnellement dans des secteurs d’activité variés intégrant l’informatique.

Les stages en milieu professionnel permettent de favoriser une meilleure adéquation entre les formations à l’Ecole et les besoins évolutifs du marché de l’emploi.

Les principaux débouchés professionnels des diplômés de l’Ecole concernent les domaines suivants :

* L’informatique de gestion d’entreprise
* Les technologies de l’information et de la communication (TIC)
* La sécurité informatique des réseaux
* L’administration des réseaux et des systèmes
* Les services bancaires et financiers, notamment le Mobile Banking
* Les télécommunications et la téléphonie mobile
* Les Big Data
* Le commerce, la vente et l’achat, le Marketing
* L’ingénierie informatique appliquée
* L’écologie et le développement durable

Parmi les sociétés, entreprises et organismes partenaires de l’Ecole, on peut citer : ACCENTURE Mauritius, Air Madagascar, Ambre Associates, Airtel, Agence Universitaire de la Francophonie (AUF) , B2B, Banque Centrale, BFG-SG, BIANCO, BLUELINE, Bureau national de gestion des Risques et des catastrophes (BNGRC), CEDII-Fianarantsoa, Data Consulting, Central Test, Centre National Antiacridien, CNRE, CHU, CNRIT, COLAS, Direction Générale des Douanes, DLC, DTS/Moov, FID, FTM, GNOSYS, IBONIA, INGENOSIA, INSTAT, IOGA, JIRAMA, JOUVE, MADADEV, MAEP, MEF, MEN, MESupRES, MFB, MIC, MNINTER, MPTDN, NEOV MAD, Ny Havana, MNP, OMNITEC, ORANGE, OTME, PRACCESS, QMM Fort-Dauphin, SMMC, SNEDADRS Antsirabe, Sénat, SEPT, SOFTWELL, Strategy Consulting, TELMA, VIVETEC, Société LAZAN’I BETSILEO, WWF …

L’organisation de stage en entreprise continue non seulement à renforcer la professionnalisation des formations dispensées, mais elle continue surtout à accroître de façon exceptionnelle les opportunités d’embauche pour les diplômés de l’Ecole.

### Partenariat au niveau international

Entre 1196 et 1999, l’ENI avait bénéficié de l’assistance technique et financière de la Mission Française de Coopération et d’action culturelle dans le cadre du Programme de Renforcement de l’Enseignement Supérieur (PRESUP) consacré à l’Ecole a notamment porté sur :

* Une dotation en logiciels, micro-ordinateurs, équipements de laboratoire de maintenance et de matériels didactiques
* La réactualisation des programmes de formation assortie du renouvellement du fonds de la bibliothèque
* L’appui à la formation des formateurs
* L’affectation à l’Ecole d’Assistants techniques français

De 2000 à 2004, l’ENI avait fait partie des membres du bureau de la Conférence Internationale des Ecoles de formation d’Ingénieurs et Technicien d’Expression Française (CITEF).

Les Enseignants-Chercheurs de l’Ecole participent régulièrement aux activités organisées dans le cadre du Colloque Africain sur la Recherche en Informatique (CARI).

L’ENI avait également signé un accord de coopération interuniversitaire avec l’Institut de Recherche en Mathématiques et Informatique Appliquées (IREMIA) de l’Université de la Réunion, l’Université de Rennes 1, l’INSA de Rennes, l’Institut National Polytechnique de Grenoble (INPG).

A partir du mois de Juillet 2001, l’ENI avait abrité le Centre de Réseau Opérationnel (Network Operating Center) du point d’accès à Internet de l’Ecole ainsi que de l’Université de Fianarantsoa. Grâce à ce projet américain qui a été financé par l’USAID Madagascar, l’ENI de l’Université de Fianarantsoa avait été dotées d’une ligne spécialisée d’accès permanent au réseau Internet.

L’ENI avait de même noué des relations de coopération avec l’Institut de Recherche pour le Développement (IRD).

L’objet du projet de coopération avait porté sur la modélisation environnementale du Corridor forestier de Fandriana jusqu’à Vondrozo (COFAV). Dans ce cadre, un atelier scientifique international avait été organisé à l’ENI en Septembre 2008. Cet atelier scientifique avait eu pour thème de modélisation des paysages.

Et dans le cadre du programme scientifique PARRUR, l’IRD avait financé depuis 2010 le projet intitulé « Forêts, Parcs et Pauvreté dans le Sud de Madagascar » (FPPSM). Des étudiants en DEA et des Doctorants issus de l’ENI avaient participé à ce Programme.

Par ailleurs, depuis toujours la même année 2010, l’ENI de Fianarantsoa avait été sélectionnée pour faire partie des organismes partenaires de l’Université de Savoie dans le cadre du projet TICEVAL relatif à la certification des compétences en TIC ;

Le projet TICEVAL avait été financé par le Fonds Francophone des Inforoutes pour la période allant de 2010 à 2012, et il avait eu pour objectif de généraliser la certification des compétences en Informatique et Internet du type C2i2e et C2imi.

Dans le cadre du projet TICEVAL, une convention de coopération avec l’Université de Savoie avait été signée par les deux parties concernées. La mise en œuvre de la Convention de Coopération avait permis d’envoyer des étudiants de l’ENI à Chambéry pour poursuivre des études supérieures en Informatique.

Enfin et non des moindres, l’ENI avait signé en Septembre 2009 un protocole de collaboration scientifique avec l’ESIROI – STIM de l’Université de la Réunion.

Comme l’ENI constitue une pépinière incubatrice de technologie de pointe, d’emplois et d’entreprises, elle peut très bien servir d’instrument efficace pour renforcer la croissance économique du pays, et pour lutter contre la Pauvreté.

De même que le statut de l’Ecole devrait permettre de renforcer la position concurrentielle de la Grande Ile sir l’orbite de la modélisation grâce au développement des nouvelles technologies.

### Débouche professionnels des diplômes

Le chômage des jeunes diplômés universitaires fait partie des maux qui gangrènent Madagascar. L’environnement socio-politique du pays depuis 2008 jusqu’ à ce jour a fait que le chômage des diplômés est devenu massif par rapport aux établissements de formation supérieure existants.

Cependant, les formations proposées par l’Ecole permettent aux diplômés d’être immédiatement opérationnels sur le marché du travail avec la connaissance d’un métier complet lié à l’informatique aux TIC.

L’Ecole apporte à ses étudiants un savoir-faire et un savoir-être qui les accompagnent tout au long de leur vie professionnelle. Elle a une vocation professionnalisante.

Les diplômés en LICENCE et en MASTER issus de l’ENI peuvent faire carrière dans différents secteurs.

L’Ecole bénéficie aujourd’hui de 34 années d’expériences pédagogiques et de reconnaissance auprès des sociétés, des entreprises et des organismes. C’est une Ecole Supérieure de référence en matière informatique.

Par conséquent, en raison de fait que l’équipe pédagogique de l’Ecole est expérimentée, les enseignants-chercheurs et les autres formateurs de l’Ecole sont dotés d’une grande expérience dans l’enseignement et dans le milieu professionnel.

L’Ecole est fière de collaborer de façon régulière avec un nombre croissant d’entreprises, de sociétés et d’organismes publics et privés à travers les stages des étudiants. Les formations dispensées à l’Ecole sont ainsi orientées vers le besoin et les attentes des entreprises et des sociétés.

L’Ecole fournit à ses étudiants de niveau LICENCE et MASTER des compétences professionnelles et métiers indispensables pour les intégrer sur le marché du travail.

L’Ecole s’efforce de proposer à ses étudiants une double compétence à la fois technologique et managériale combinant l’informatique de gestion ainsi que l’administration des réseaux et systèmes.

D’une manière générale, les diplômés de l’ENI n’éprouvent pas de difficultés particulières à être recrutés au terme de leurs études. Cependant, l’ENI recommande à ses diplômés de promouvoir l’entrepreneuriat en TIC et de créer des cybercafés, des SSII ou des bureaux d’études.

Tableau 3 montre les débouchés professionnels éventuels des diplômés.

Tableau 3: Débouchés professionnels éventuels des diplômés

|  |  |
| --- | --- |
| LICENCE | * Analyste * Programmeur * Administrateur de site web/de portail web * Assistant Informatique et internet * Chef de projet web ou multimédia * Agent de référencement * Technicien/Supérieur de help desk sur Informatique * Responsable de sécurité web * Administrateur de réseau |
| MASTER | * Administrateur de cybercafé * Administrateur de réseau et système * Architecture de système d’information * Développeur d’application /web /java/Python/ IOS /Android * Ingénieur réseau * Webmaster /web designer * Concepteur Réalisateur d’applications * Directeur du système de formation * Directeur de projet informatique * Chef de projet informatique * Responsable de sécurité informatique * Consultant fonctionnel ou freelance * Administrateur de cybercafé |

### Ressources humaines

* Directeur de l’Ecole : Professeur RAMAMONJISOA Bertin Olivier
* Responsable de Mention : Docteur MAHATODY Thomas
* Responsable de Parcours « Génie Logiciel et Base de Données » : Docteur RATIARSON Venot
* Responsable de Parcours « Administration Systèmes et Réseaux » : Monsieur SIAKA
* Responsable de Parcours « Informatique Générale » : Docteur RAKOTOASIMBAHOAKA Cyprien Robert
* Nombre d’Enseignants permanents : 13 dont deux (02) Professeurs Titulaires, six (06) Maîtres de Conférences et cinq (05) Assistants d’Assistants d’Enseignement Supérieur et de Recherche
* Nombre d’Enseignants vacataires : 10
* Personnel Administratif : 23

## Chapitre 2 : Présentation de LIVDHASBJF

### 2.1Historique et missions

Livenexx Software est une Entreprise de Société Numérique (ESN) et mobile multiculturelle composées d’une quinzaine de collaborateurs. Crée en Juin 2016 par Monsieur Richad HASSIM, elle a pour missions de créer des sites vitrines entièrement personnalisés, des sites E-commerce, des applications web et mobiles reflétant les organisations des clients. Elle se situe à Antsakaviro, Rue Razafindralambo Pierre, Antananarivo.

Site Web : [www.livenexx.fr](http://www.livenexx.fr/)

Adresse électronique : commercial@livenexx.fr

Téléphone : +261 32 05 326 08.

L’agence se démarque par le fait qu’elle offre :

* Des solutions sur mesure : Conceptions et techniques adaptées aux activités des clients ainsi qu’à leur identité de marque pour une solution unique.
* Des solutions flexibles : Panel de technologies permettant de s’adapter à tous les budgets.
* Des produits de qualité : Organisation du travail respectant les standards de l’industrie en matière de livraison.
* Des supports ultras réactifs : support multicanal répondant rapidement pour que les clients puissent se concentrer sur l’essentiel.
* UX Design : Travail suivant constamment les nouvelles tendances et assurant l’ergonomie sur tous les supports.
* SAV : Garanties 3 mois après livraison avec un support 100% gratuit en tout temps, de sorte que le client puisse pleinement prendre en main son système d’information.

Marketing intégré : outils d’analyse statistique et d’audience afin de faire décoller le business des clients.

### 2.2. Organisation générale

La figure 2 représente l’organigramme de la société Livenexx.

DIRECTEUR GENERAL

DIRECTEUR TECHNIQUE

CHEF DE POLE

WEB

EQUIPE DE DEVELOPPEMENT

CHEF DE POLE

CMS

CHEF DE POLE

MOBILE

RESPONSABLE

ASSURANCE ET QUALITE

RESPONSABLE

ADMINISTRATIF ET

FINANCIER

RESPONSABLE

CLIENTELE

Figure 2: Organigramme de Livenexx

L’organisation générale de la société Livenexx est fixée comme suit :

* La Direction générale définit et pilote la stratégie de l’entreprise.
* Le Responsable Administratif et Financier assure le recrutement et la gestion du personnel en fonction de la stratégie de l’entreprise. Il détermine les budgets prévisionnels correspondants.
* La Direction Technique a pour mission de gérer l’aspect technique et budgétaire d’un projet en veillant aux risques, aux coûts cachés et aux retours sur investissement.
* Le Responsable de l’Assurance Qualité définit et coordonne la mise en œuvre de la politique d’assurance qualité c’est-à-dire qu’il veille au respect des normes et des règles de sécurité et en garantit l’application.
* Le Responsable clientèle est chargée de prendre contact avec les clients et s'assure de répondre à toutes leurs questions.
* Les chefs de pôles coordonnent l’ensemble des travaux de l’équipe de développement.

**2.3. Processus de développement**

La figure 3 représente le processus de développement de la société LIVENEXX.



Figure 3 : Processus de développement de la société LIVENEXX

## Chapitre 3 : Description du projet

### 3.1. Formulation

L’entreprise AKADIN ne dispose pas encore une plateforme web permettant de gérer les parcs immobiliers. Ainsi, il a mandaté la société LIVENEXX pour concevoir et réaliser une plateforme web pour gérer les parcs immobiliers.

**3.2. Objectifs et besoins de l’utilisateur**

Ce projet a pour but de faciliter la gestion de parc immobilier pour l’entreprise AKADIN.

Les utilisateurs ont besoin de :

* Suivre les statistiques du rapport entre les bénéfices et les dépenses, ainsi que les chiffres d’affaires.
* Gérer les entités
* Gérer les biens
* Gérer les locataires des biens
* Gérer les baux
* Gérer les facturations
* Gérer la ressources humaines
* Gérer les dépenses
* Gérer les fournisseurs
* Gérer les stocks
* Gérer les caisses
* Gérer les rentabilités
* Gérer les taxes
* Gérer les projets
* Gérer les alertes
* Gérer les TVA
* Gérer les contrats
* Gérer les rapports journaliers, mensuels, annuels

**3.3. Moyens nécessaires à la réalisation du projet**

Pour la mise en œuvre du projet, des moyens humains, matériels et logiciels se sont complétés.

* Moyens humains
* Concepteur : fait la conception du projet
* Chef de projet : encadre le déroulement du projet
* Développeurs Frontend : développent la partie cote utilisateur
* Développeurs Backend : développent la partie cote serveur
* Testeurs : testent l’application
* Moyens matériels

Tableau 4: Moyens matériels à la réalisation du projet

|  |  |  |  |
| --- | --- | --- | --- |
| Désignation | Marque | Caractéristiques | Système d’exploitation |
| PC Portable | Acer | HDD : 500 Go  RAM : 4Go  CPU : Corei5 | Windows |

* Moyens logiciels
  + Skype : Outil de communication
  + Google Chrome : outil de visualisation de résultat
  + PhpStorm : environment de devéloppement
  + Thunderbird : outil pour envoyer un rapport de travail fait journalièrement
  + Xampp : serveur MySQL
  + Visual Paradigm : outil de modelisation

# ANALYSE ET CONCEPTION

## Chapitre 4. Analyse préalable

### 4.1. Analyse de l’existant

### 4.1.1. Organisation actuelle

L’entreprise AKADIN

### 4.1.2. Inventaire des moyens matériels et logiciels

* Moyens matériels

Le tableau 5 montre l’inventaire des moyens matériels

Tableau 5: Inventaire des moyens matériels

|  |  |  |
| --- | --- | --- |
| Désignation | Caractéristique | Système d’exploitation |
| Ordinateur Acer | RAM : 4 Go  HDD : 500 Go  CPU : Corei5 | Windows 10 |
| Ordinateur Asus | RAM : 6 Go  HDD : 500 Go  CPU : Corei5 | Windows 10 |
| Ordinateur Toshiba | RAM : 8 Go  HDD : 500 Go  CPU : Corei5 | Windows 10 |
| Ordinateur Asus | RAM : 8 Go  HDD : 1 To  CPU : Corei5 | Windows 10 |
| Ordinateur Lenovo | RAM : 4 Go  HDD : 500 Go  CPU : Corei5 | Windows 10 |
| Ordinateur Hp | RAM : 4 Go  HDD : 500 Go  CPU : Corei5 | Windows 10 |
| Ordinateur MacBook | RAM : 8 Go  HDD : 500 Go  CPU : Corei5 | MacIntosh |

* Moyens logiciels
* Navigateur: Google Chrome
* Skype
* ThinderBird
* Systeme d’exploitation: Windows

### 4.2. Critique de l'existant

* **Points Forts**
* Pas de perte d’information.
* **Points Faibles**
* La recherche d’information demande beaucoup de temps.
* Perte d’information.

### 4.3. Conception avant-projet

### 4.3.1. Proposition des solutions

* Acheter une plateforme pour la gestion de parc immobilier.

**Avantages**

1. Gain de temps : c'est à dire que la mise en place de l’application ne sera pas longue.
2. Outil robuste et opérationnel : ces logiciels sont prêts à être utilisés et offre indéniablement des fonctions répondant aux besoins de ses utilisateurs.

**Inconvénients**

1. Coût élevé : les versions complètes des logiciels sont généralement onéreuses.
2. Non concordance des services : en effet, le produit ne s'harmonise jamais parfaitement aux besoins de l’établissement.
3. Problème d’agencement : si le code source du programme n'est pas accessible (ce qui est généralement le cas), il ne peut y avoir d'amélioration vis à vis des penchants de l’établissement.

* Créer une plateforme web pour la gestion de parc immobilier.

**Avantages**

1. Possibilité d'extensions et amélioration de la plateforme.
2. Facilité de maintenance.

**Inconvénient**

1. Le développement d'une plateforme implique beaucoup de temps et de travails.

### 4.3.2. Solution retenue

La conception et la réalisation de la plateforme web sur la « gestion de parc immobilier » a été la solution retenue après l’analyse approfondie des coûts et bénéfices.

Pour réaliser cette solution, nous avons besoin d’une méthode de conception, un langage de modélisation, un outil de conception, un langage de programmation, un système de gestion de base de données, un environnement de développement.

**4.3.2. Choix de la méthode de Conception**

Une méthode de conception est un procédé permettant de construire un modèle aussi correct que possible et aussi efficacement que possible. Une méthode d’analyse et de conception a donc pour objectif de permettre de formaliser les étapes préliminaires du développement d’un système d’information afin de rendre ce développement plus fidèle aux besoins du client. [2]

Le tableau 6 va illustrer une comparaison entre deux méthodes de conception les plus utilisées.

Tableau 6: Comparaison entre deux méthodes de conception

|  |  |  |
| --- | --- | --- |
| Méthodes | Avantages | Inconvénients |
| 2TUP | * Processus orienté objet * Itérative * Fait une large place à la technologie et à la gestion des risques | * Plutôt superficielle sur les phases situées en amont et en aval du développement : capture des besoins, support, maintenance, gestion du changement |
| MERISE | * Méthode orientée fonction * Considère le système réel selon deux points de vue : * Vue statique (données) * Vue dynamique (traitements) | * Non orienté objet |

Le choix s’est porté sur 2TUP, qui est un processus orienté objet c’est-à-dire qu’il permet d’encadrer la production d’artefacts (diagrammes et autres modèles) produit par un langage de modélisation et de conception orienté objet UML.

Le processus de développement du projet passe par la pré-étude qui fait sortir les fonctionnalités du système, aussi par l’élaboration de l’architecture technique ainsi que le codage final. La méthode UP (Processus Unifié) est générique c’est-à-dire qu’elle définit un certain nombre critères de développement, chaque société peut par la suite personnaliser afin de créer son propre processus adapté à ses besoins. C’est dans ce cadre que la société Valtech a créé la méthode 2TUP qui signifie « 2 Track Unified Process ». C’est un processus qui répond aux caractéristiques du Processus Unifié.

Un processus définit une séquence d’étapes ordonnées qui permet de produire un système logiciel ou de faire évoluer un système existant.

2TUP a pour but d’apporter une réponse aux contraintes de changement fonctionnelles et techniques qui s’imposent aux systèmes d’information. Il propose un cycle de développement qui dissocie les aspects techniques des aspects fonctionnels. Il distingue ainsi deux branches dont les résultats sont fusionnés pour réaliser le système.

On obtient donc un processus de développement en Y :

* **La branche fonctionnelle (gauche) :**  vise l’analyse des spécifications fonctionnelles de manière à déterminer ce que va réaliser le système en termes de métier. C’est ici qu’on identifie et dégage toutes les fonctionnalités du système à réaliser.
* **La branche technique (droite) :**  recense toutes les contraintes et les choix dimensionnant la conception du système. Puis, les techniques développées pour le système peuvent l’être en effet indépendamment des fonctions à réaliser.
* **La phase de réalisation :**  Cette phase est la fusion des deux précédentes et mène à la conception applicative et à la solution adaptée aux besoins des utilisateurs. Elle concerne les étapes de la conception préliminaire, la conception détaillée, le codage et les tests puis l’étape de recette et de déploiement.

La figure 4 illustre le cycle de développement en Y de 2TUP.

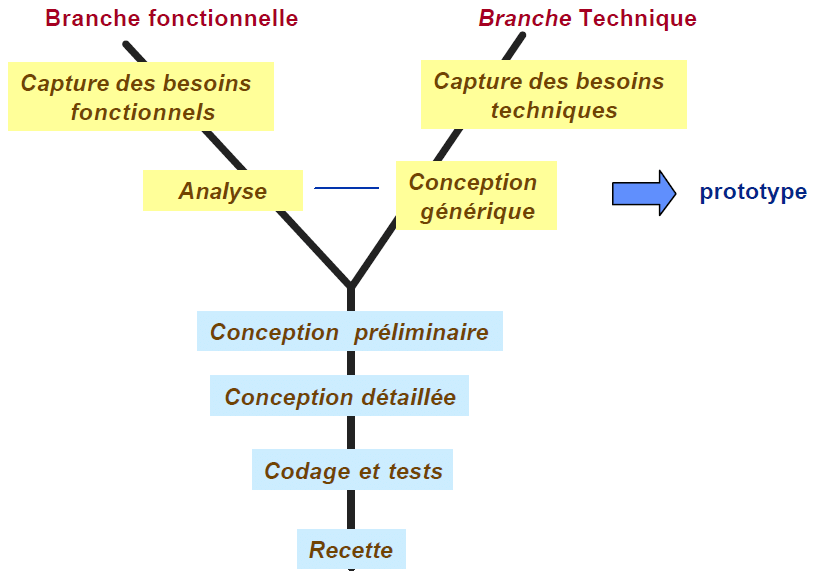


Figure 4: Cycle de développement en Y de 2TUP

Certaines méthodes sont associées à un langage de modélisation. D’ailleurs, la méthode de modélisation 2TUP est associée au langage de modélisation UML.

Un langage de modélisation est une syntaxe commune, graphique, pour modéliser (OMT, UML).

**Présentation d’UML [2]**

* UML (Unified Modeling Language) est un langage de modélisation graphique à base de pictogramme.
* Il est conçu pour représenter, construire et documenter des systèmes logiciels utilisant les techniques orientées objet.
* Il permet la création de plusieurs modèles d’un même système, chacun privilégiant un aspect différent : fonctionnel, dynamique, statique.
* Il offre plusieurs niveaux d’abstraction qui simplifient la conception des solutions.
* Sa notation graphique est très adaptée à la conception orientée objet, et sera beaucoup plus simple à manipuler que l’algorithme classique.
* Son côté visuel facilite également la comparaisonetl’évaluationdesolutions.
* C’est le résultat d’un large consensus et du travail d’experts reconnus.
* UML est ainsi devenu un langage incontournabledans tout projet de taille conséquente.

UML dans sa version 2 propose treize diagrammes qui peuvent être utilisés dans la description d’un système. Ces diagrammes sont regroupés dans deux grands ensembles :

* **Les diagrammes structurels :** Ces diagrammes, au nombre de six, ont vocation à représenter l’aspect statique d’un système (classes, objets, composants...).

Le tableau 7 illustre les différents diagrammes structurels.

Tableau 7: Diagramme Structurel

|  |  |
| --- | --- |
| **Diagramme**  **Structurel** | **Description** |
| **Diagramme de**  **Classe** | * Diagramme pivot de l’ensemble de la modélisation d’un système. * Représente la description statique du système en intégrant dans chaque classe la partie dédiée aux données et celle consacrée aux traitements. |
| **Diagramme d’Objet** | * Permet la représentation d’instances des classes et des liens entre instances. |
| **Diagramme de**  **Composant** | * Représente les différents constituants du logiciel au niveau de l’implémentation d’un système. |
| **Diagramme de**  **Déploiement** | * Ce diagramme décrit l’architecture technique d’un système avec une vue centrée sur la répartition des composants dans la configuration d’exploitation. |
| **Diagramme de**  **Paquetage** | * Donne une vue d’ensemble du système structuré en paquetage. Chaque paquetage représente un ensemble homogène d’éléments du système (classes, composants...). |
| **Diagramme de**  **Structure Composite** | * Permet de décrire la structure interne d’un ensemble complexe composé par exemple de classes ou d’objets et de composants techniques. |

* **Les diagrammes de comportement :** Ces diagrammes représentent la partie dynamique d’un système réagissant aux évènements et permettant de produire les résultats attendus par les utilisateurs. Sept diagrammes sont proposés par UML :

Le tableau 8 Illustre la description des différents diagrammes de comportement.

Tableau 8: Diagramme de comportement

|  |  |
| --- | --- |
| **Diagramme de**  **Comportement** | **Description** |
| **Diagramme de Cas**  **D’Utilisation** | * Un des diagrammes les plus structurants dans l’analyse d’un système. * Représente les besoins des utilisateurs par rapport au système. |
| **Diagramme d’Etat-**  **Transition** | * Montre les différents états des objets en réaction aux évènements. |
| **Diagramme**  **D’Activité** | * Donne une vision des enchaînements des activités propres à une opération ou à un cas d’utilisation. |
| **Diagramme de**  **Séquence** | * Décrit les scénarios de chaque cas d’utilisation en mettant l’accent sur la chronologie des opérations en interaction avec les objets. |
| **Diagramme de**  **Communication** | * Représente des scénarios des cas d’utilisation qui met plus l’accent sur les objets et les messages échangés. |
| **Diagramme Global**  **D’Interaction** | * Fournit une vue générale des interactions décrites dans le diagramme de séquence et des flots de contrôle décrits dans le diagramme d’activités. |
| **Diagramme de**  **Temps** | * Représente les états et les interactions d’objets dans un contexte où le temps a une forte influence sur le comportement du système à gérer. |

1. **Choix de l’outil de conception**

Modéliser un système consiste à créer une représentation virtuelle d’une réalité de façon à faire ressortir les points auxquels on s’intéresse.

Les outils de conception aident à concevoir, simuler et réaliser des produits de meilleure qualité en réduisant le coût de développement. Ils sont variés.

Le tableau 9 illustre une comparaison entre deux types d’outils de conception.

Tableau 9: Comparatif d'outils de conception

|  |  |  |
| --- | --- | --- |
|  | VISUAL PARADIGM FOR UML | UML DESIGNER |
| Caractéristiques | * Libre * Multiplateforme * Différentes fonctions facilitant la conception des diagrammes * Génération de code source à partir des diagrammes * Réalisation du mapping object automatiquement à partir des diagrammes de classe | * Open source * Multiplateforme * Intégrable dans Eclipse * Possibilité de créer ses propres diagrammes |

Visual Paradigm For Uml est l’outil de conception choisi car il représente des multiples avantages et il est facile à manipuler. Visual Paradigm permet la création des diagrammes UML et des modèles qui en sont à l’origine.

Il peut alors générer du code dans un langage de programmation déterminé. Il propose également la création d’autres types de diagrammes, comme celui qui permet la modélisation des bases de données. Et en plus il permet de générer du code SQL qu’il peut ensuite déployer automatiquement dans différents environnements. [2]

Le plus important c’est que Visual Paradigm peut générer la base de données et les codes de mise à jour de la base de données (setters et getters).

### Choix du langage de Programmation

En informatique, un langage de programmation est une notation conventionnelle destinée à formuler des algorithmes et produire des programmes informatiques qui les appliquent. D’une manière similaire à une langue naturelle, un langage de programmation est composé d’un alphabet, d’un vocabulaire, des règles de grammaire et de significations.

Il permet de décrire d’une part les structures des données qui seront manipulées par l’appareil informatique, et d’autre part d’indiquer comment sont effectuées les manipulations, selon quels algorithmes. Ils servent de moyens de communication par lesquels le programmeur communique avec l’ordinateur, mais aussi avec d’autres programmeurs.

Le tableau 10 illustre quelques comparatifs de langages de programmations.

Tableau 10: Comparatifs des langages de programmation

|  |  |  |
| --- | --- | --- |
| **LANGAGE** | **Avantages** | **Inconvénients** |
| **PHP** | * Prise en main rapide (utilise son propre espace mémoire) * Open source * Simplicité d’écriture de script * Facilité d’hébergement * Documentation très fournie | * Langage non typé * Des failles de sécurité * Problèmes dues à la gestion automatique des types de données |
| **JAVA** | * Orienté Objet * Multiplateforme grâce à la machine virtuelle * Communauté active * Complet, facile d’apprentissage | * Lourde * Manque de souplesse dans la syntaxe * Peu de nouveauté à attendre |

D’après le tableau de comparaison ci-dessus le choix retenu est le langage PHP pour sa facilité de prise en main rapide et sa simplicité au niveau du développement.

Présentation de PHP

Le langage PHP (HyperText Preprocessor) est un langage apparu en 1994, c’est un langage de programmation orienté objet comme Le C++. PHP est un langage de programmation interprété côté serveur : dans cette architecte, le serveur interprète le code PHP des pages web demandées et génère du code (HTML, XHTML, CSS par exemple) et les données (JPEG, GIF, PNG par exemple) pouvant être interprétés et rendus par un navigateur. [5]

Il a été conçu pour permettre la création d’application dynamique, le plus souvent développées pour le Web. PHP est le plus souvent couplé à un serveur Apache bien qu’il puisse être installé sur la plupart des serveurs HTTP.

Il est multiplateforme : autant sur Linux qu’avec Windows. Il permet aisément de reconduire le même code sur un environnement à peu près semblable (prendre en compte les règles d’arborescences de répertoires qui peuvent changer). [5]

Le langage PHP a été choisi pour développer l’application vu qu’il est le plus apte à répondre aux besoins du présent projet, si l’on ne citait que ses richesses fonctionnelles et c’est l’un des langages de programmations web le plus utilisé au niveau des grandes entreprises telles que : Facebook, Amazon, Wikipédia, YouTube, etc. ...

### Choix du Système de Gestion de Base de Données (SGBD)

Un SGBD ou Système de Gestion de Base de Données est un logiciel système destiné à stocker et à partager des informations dans une base de données, en garantissant la qualité, la pérennité et la confidentialité des informations, tout en cachant la complexité des opérations. [6]

Ces opérations consistent surtout à enregistrer des données, les rechercher, les modifier et puis surtout de créer automatiquement des comptes rendus du contenu de la base de données. Il permet de spécifier les types de données, la structure des données contenues dans la base de données, ainsi que des règles de cohérence telles que l’absence de redondance.

QuelquesSGBD feront l’objet d’une comparaison dans l’ultime but d’en choisir un qui conviendra à la réalisation du projet.

Le tableau 11 représente les comparatifs de quelques SGBD.

Tableau 11: Comparatifs des SGBD

|  |  |  |
| --- | --- | --- |
| **SGBD** | **Avantages** | **Inconvénients** |
| **MySQL** | * Hébergement public * Open Source * Facilité de déploiement * Très bonne intégration dans l’environnement Apache/PHP * Simple et Fiable * Facilité d’administration avec PhpMyAdmin | * Logiciel Propriétaire * Assez peu de richesse fonctionnelle * En cas de corruption, peu d’outils aident à la réparation * Pas de vue matérialisée |
| **PostGreSQL** | * Fiable et performant * Gratuit et Open Source * Sécurisé * Très riche fonctionnellement * Simple d’utilisation et administration * Héritage de table | * Les compétences à acquérir et maintenir * Charge de travail supplémentaire pour l’exploitation * Pas de requêtes récursives * Pas de services Web * Pas possible de requêter sur plusieurs bases à la fois |

Le SGBD MySQL a été choisi dans le développement de ce projet grâce aux nombreux avantages qu’il procure s’accordant aux normes de la Société.

MySQL est un système de gestion de base de données qui fait partie des logiciels de gestion de base de données les plus utilisés dans le monde, autant par le grand public que par des professionnels. Il est davantage orienté vers le service de données déjà en place que vers celui de la mise à jour fréquente et fortement sécurisées. Il est en effet multithread et multi-utilisateur. [7]

MySQL fonctionne sur de nombreux systèmes d’exploitation tels que Linux, Windows et Mac OS X.

Le couple PHP/MySQL est très utilisé par les sites web et proposé par la majorité des hébergeurs Web. Plus de la moitié des sites Web fonctionnent sous Apache, qui est le plus souvent utilisé conjointement avec PHP/MySQL. [7]

### Choix du Framework PHP

L’utilisation d’un Framework présente plusieurs avantages : l’accélération du temps de développement, la conservation d’un code homogène et une pérennité certaine des projets.

Enfin, les applications sont facilement sécurisées. Par conséquent, les Framework sont en général mieux conçus et plus robustes qu’une application développée par un seul programmeur. Un nombre important de Framework est disponible dans de nombreux langages. Il n’y a pas de Framework idéal, chacun possède ses avantages et ses inconvénients. Pour faire le bon choix il convient, après une première présélection, d’en tester plusieurs afin de déterminer celui répondant mieux à vos besoins.

Parmi la multitude d’excellents Framework PHP, nous avons choisi de comparer trois d’entre eux, adoptant tous la structure MVC (Model-View-Controller).

Le tableau 12 montre la comparaison de ces Framework PHP

Tableau 12 : Comparaison de ces Framework PHP

|  |  |  |
| --- | --- | --- |
| Framework | Avantage | Inconvénients |
| Symfony | * Incontournable dans le   monde professionnel   * Robuste et fiable * Sécurisé | * Temps d’apprentissage   considérable |
| Laravel | * Communauté active * Syntaxe Orienté Objet * Rapide, léger | * Framework jeune * Nécessite d’examiner une documentation complexe |
| CodeIgniter | - Documentation  complète  - Prise en main rapide  - Simple, léger | - Nécessite plus de lignes  de codes   * Pas de moteur de template |

Les avantages et inconvénients n’ont pas été assez discriminants pour permettre de choisir un Framework parmi tous les autres. Ainsi, pour choisir celui qui serait le plus intéressant à utiliser, je me suis concentré sur une solution puissante et à une très grande performance qui répond à toutes les exigences de notre projet. Ainsi, mon choix s’est porté sur Symfony. En effet, celui-ci est un Framework PHP très populaire en entreprise et réputé pour sa robustesse, sa flexibilité et qui mise surtout sur la sécurisation.

Le développement sur Symfony se fait par le langage de programmation PHP, Etant un langage orienté objet, interprété et à dynamique, il offre une facilité d’utilisation et de documentation avec sa communauté très active.

Chapitre 5 : Analyse conceptuelle

5.1. Dictionnaire de données

Le tableau 13 montre le dictionnaire de données

Tableau 13 : Dictionnaire de données

|  |  |  |  |
| --- | --- | --- | --- |
| Nom | Description | Structure | |
| Type | Longueur |
| lpi\_bank\_id | Identifiant de la banque |  |  |
| bk\_label | Libelle de la banque |  |  |
| bk\_img\_url | Url de l’image de la banque |  |  |
| lpi\_country\_id | Identifiant du pays |  |  |
| cntr\_label | Libelle du pays |  |  |
| lpi\_currency\_id | Identifiant de la devise |  |  |
| cur\_label | Libelle de la devise |  |  |
| cur\_is\_current | Devise actuel |  |  |
| lpi\_property\_id | Identifiant du bien |  |  |
| ppt\_name | Nom de la propriété |  |  |
| ppt\_building\_plan | Plan du bâtiment |  |  |
| ppt\_is\_electrified | Accès électricité |  |  |
| ppt\_num | Numéro de la propriété |  |  |
| ppt\_date\_finalized | Date de finalisation |  |  |
| lpi\_payment\_type\_id | Identifiant type de paiement |  |  |
| pmt\_tp\_label | Libelle du type de paiement |  |  |
| lpi\_property\_status\_id | Identifiant du statut du bien |  |  |
| ppt\_st\_label | Libelle du statut du bien |  |  |
| lpi\_field\_id | Identifiant du terrain |  |  |
| fld\_label | Nom du terrain |  |  |
| fld\_is\_built | Le terrain est bâti |  |  |
| fld\_area | Superficie du terrain |  |  |
| fld\_longitude | Longitude du terrain |  |  |
| fld\_latitude | Latitude du terrain |  |  |
| fld\_address | Addresse du terrain |  |  |
| fld\_num | Numéro du terrain |  |  |
| lpi\_entity\_id | Identifiant de l’entité |  |  |
| ent\_label | Nom de l’entité |  |  |
| ent\_manager\_name | Nom du gérant de l’entité |  |  |
| ent\_tax\_coordinate | Coordonne de la taxe de l’entité |  |  |
| ent\_number | Numéro de l’entité |  |  |
| lpi\_agency\_id | Identifiant de l’agence |  |  |
| agc\_label | Nom de l’agence |  |  |
| agc\_address | Addresse de l’agence |  |  |
| agc\_tel | Numéro téléphone de l’agence |  |  |
| agc\_agent\_name | Nom de l’agent |  |  |
| ctr\_date\_created | Date de création du contrat |  |  |
| ctr\_date\_effective | Date d’entrée en vigueur |  |  |
| ctr\_payment\_mensual\_first\_date | Date de première mensualité |  |  |
| ctr\_num | N° de contrat |  |  |
| ctr\_warranty\_desc | Description du garanti |  |  |
| ctr\_warranty\_amount | Montant de la garanti |  |  |
| ctr\_payment\_number | Nombre de versement par an |  |  |
| ctr\_payment\_mensual\_planned | Nombre de mensualités prévues |  |  |
| ctr\_amount | Montant du contrat |  |  |
| ctr\_duration | Durée du contrat |  |  |
| ctr\_mensual\_amount | Montant mensuel du contrat |  |  |
| ctr\_benefit\_amount | Montant bénéficié du contrat |  |  |
| ctr\_annual\_benefit\_rate | Taux de benefice annuel du contrat |  |  |
| ctr\_additionnal\_amount | Montant additionnel du contrat |  |  |

5.2. Règle de gestion

RG1 : La date de la première mensualité doit être supérieure à la date d’entrée en vigueur.

RG2 : Si le contrat est garanti, il y a une description.

RG3 : Si le contrat n’est pas garanti, un bien sera pris

Chapitre 6 : Analyse conceptuelle

6.1. Diagramme de classe de conception

### 6.1.1 Identification des acteurs

Un acteur représente l’abstraction d’un rôle joué par des entités externes (utilisateur, dispositif matériel ou autre système) qui interagissent directement avec le système étudié. Un acteur peut consulter et modifier directement l’état du système, en émettant et/ou en recevant des messages éventuellement porteurs de données.

La figure 5 représente l’acteur qui est correspond à un stick man.

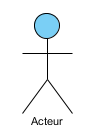


Figure 5: Acteur

Le tableau 15 représente les acteurs interagissant avec le système étudié avec leurs descriptions respectives.

Identification des acteurs avec leurs descriptions

|  |  |
| --- | --- |
| Acteur | Description |
| Superadmin | * Personne habilitée à gérer la plateforme de gestion de parc immobilier |

### 6.2 Spécification des besoins fonctionnels

#### 6.2.1 Identification des cas d’utilisation

Le tableau 18 représente l’identification des cas d’utilisation.

|  |  |  |
| --- | --- | --- |
| Cas d’Utilisation | Acteur | Message Emis / Reçu |
| Gérer les utilisateurs | Superadmin | Demande d’ajout d’un utilisateur |
| Résultat de la demande d’ajout d’un utilisateur |
| Demande de modification d’un utilisateur |
| Résultat de la demande de modification d’un utilisateur |
| Demande de suppression d’un utilisateur |
| Résultat de la demande de suppression d’un utilisateur |
| Gérer les pays | Superadmin | Demande d’ajout d’un pays |
| Résultat de la demande d’ajout d’un pays |
| Demande de modification d’un pays |
| Résultat de la demande de modification d’un pays |
| Demande de suppression d’un un pays |
| Résultat de la demande de suppression d’un pays |
| Gérer les banques | Superadmin | Demande d’ajout d’une banque |
| Résultat de la demande d’ajout d’une banque |
| Demande de modification d’une banque |
| Résultat de la demande de modification d’une banque |
| Demande de suppression d’une banque |
| Résultat de la demande de suppression d’une banque |

|  |  |  |
| --- | --- | --- |
| Gérer les contrats | Superadmin | Demande d’ajouter un contrat |
| Résultat de la demande d’ajout d’un contrat |
| Demande de modifier un contrat |
| Résultat de la demande de modification d’un contrat |
| Demande de supprimer un contrat |
| Résultat de la demande de suppression d’un contrat |
| Gérer les devises | Superadmin | Demande d’ajouter une devise |
| Résultat de la demande d’ajout d’une devise |
| Demande de modifier une devise |
| Résultat de la demande de modification d’une devise |
| Demande de supprimer une devise |
| Résultat de la demande de suppression d’une devise |
| Gérer les types de paiements | Superadmin | Demande d’ajouter un type de paiement |
| Résultat de la demande d’ajout d’un type de paiement |
| Demande de modifier un type de paiement |
| Résultat de la demande de modification d’un type de paiement |
| Demande de supprimer un type de paiement |
| Résultat de la demande de suppression d’un type de paiement |

|  |  |  |
| --- | --- | --- |
| Gérer les statuts de bien | Superadmin | Demande d’ajouter un statut de bien |
| Résultat de la demande d’ajout d’un statut de bien |
| Demande de modifier un statut de bien |
| Résultat de la demande de modification d’un statut de bien |
| Demande de supprimer un statut de bien |
| Résultat de la demande de suppression d’un statut de bien |
| Gérer les biens | Superadmin | Demande d’ajouter un bien |
| Résultat de la demande d’ajout d’un statut de bien |
| Demande de modifier un bien |
| Résultat de la demande de modification d’un bien |
| Demande de supprimer un bien |
| Résultat de la demande de suppression d’un bien |

|  |  |  |
| --- | --- | --- |
| Cas d’Utilisation | Acteur | Message Emis / Reçu |
| Gérer les entités | Superadmin | Demande d’ajouter une entité |
| Résultat de la demande d’ajout d’une entité |
| Demande de modification d’une entité |
| Résultat de la demande de modification d’une entité |
| Demande de suppression d’une entité |
| Résultat de la demande de suppression d’une entité |
| Gérer les terrains | Superadmin | Demande d’ajouter un terrain |
| Résultat de la demande d’ajout d’un terrain |
| Demande de modification d’un terrain |
| Résultat de la demande de modification d’un terrain |
| Demande de suppression d’un terrain |
| Résultat de la demande de suppression d’un terrain |
| Gérer les agences | Superadmin | Demande d’ajout d’une agence |
| Résultat de la demande d’ajout d’une agence |
| Demande de modification d’une agence |
| Résultat de la demande de modification d’une agence |
| Demande de suppression d’une banque |
| Résultat de la demande de suppression d’une agence |

#### 6.3 Diagramme de cas d’utilisation

Les cas d’utilisation décrivent le comportement du système du point de vue de l’utilisateur. Ils permettent de définir les limites du système et les relations entre le système et son environnement. Un cas d’utilisation est une manière spécifique d’utiliser le système. C’est l’image d’une fonctionnalité en réponse à la stimulation d’un acteur externe. Les composants d’un diagramme de cas d’utilisation sont :

* **L’acteur :** un acteur est celui qui interagie directement avec le système.
* **User Case :** un cas d’utilisation (« use case ») représente un ensemble de séquences d’actions qui sont réalisées par le système et qui produisent un résultat observable intéressant pour un acteur particulier. Un cas d’utilisation doit être relié à au moins un acteur.

Un cas d’utilisation est montré dans la figure 6.

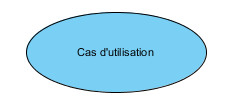


Figure 6 : Cas d'Utilisation

* **Association :** Utilisé pour relier des acteurs et des cas d’utilisation. Elles indiquent qu’un acteur participe au cas d’utilisation sous une forme quelconque. Les associations sont représentées par une ligne reliant l’acteur et le cas d’utilisation.

La figure 7 montre comment ces trois éléments de base collaborent pour former un diagramme de cas d’utilisation.

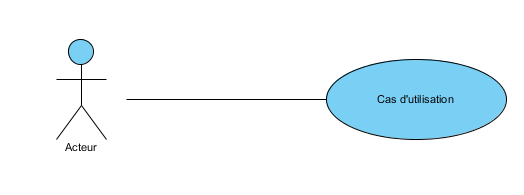


Figure 7 : Formalisme d'un diagramme de cas d'utilisation

* **Relation**
* **Include :** Relation spécifiant qu’un use case en incorpore explicitement et obligatoirement un autre en un point donné de ses enchaînements.

La figure 8 représente le formalisme d’une inclusion.

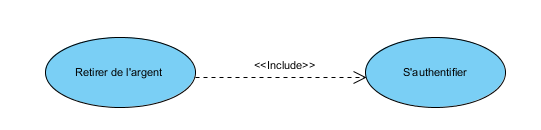


Figure 8 : Formalisme d'une inclusion

* **Extend :** Relation spécifiant qu’un use case peut optionnellement en incorporer un autre en un point donné de ses enchaînements.

La figure 9 représente le formaliste d’une extension.

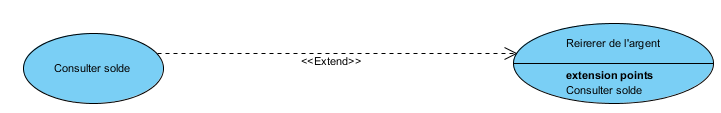


Figure 9 : Formalisme d'une extension

La figure 10 illustre le diagramme de cas d’utilisation.



Figure 10 : Diagramme de cas d’utilisation

#### 6.4 Diagramme de séquence système

Les diagrammes de séquences sont la représentation graphique des interactions entre les acteurs et le système en indiquant la chronologie des échanges. Cette représentation peut se réaliser par cas d’utilisation en considérant les différents scénarios associés.

Il fait ressortir :

* Les acteurs : représentés par un bonhomme et une ligne verticale (ligne de vie).
* Les objets : représentés par un rectangle et une ligne verticale (ligne de vie de l’objet).
* Les messages : traduisent les interactions (échange d’informations) entre objets. Ils sont représentés par des flèches orientées de l’émetteur au récepteur.

Deux types de messages peuvent être distingués :

* **Message synchrone :** Dans ce cas l’émetteur reste en attente de la réponse à son message avant de poursuivre ses actions. La flèche avec extrémité pleine symbolise ce type de message. Le message retour peut ne pas être représenté car il est inclus dans la fin d’exécution de l’opération de l’objet destinataire du message.
* **Message asynchrone :** Dans ce cas, l’émetteur n’attend pas la réponse à son message, il poursuit l’exécution de ses opérations. C’est une flèche avec une extrémité non pleine qui symbolise ce type de message.

La figure 11 représente le diagramme de séquence pour le cas d’utilisation « Gérer les banques »



Figure 11: Diagramme de séquence pour le cas d’utilisation « Gérer les banques »

La figure 12 représente le diagramme de séquence pour le cas d’utilisation « Gérer les types de paiements ».

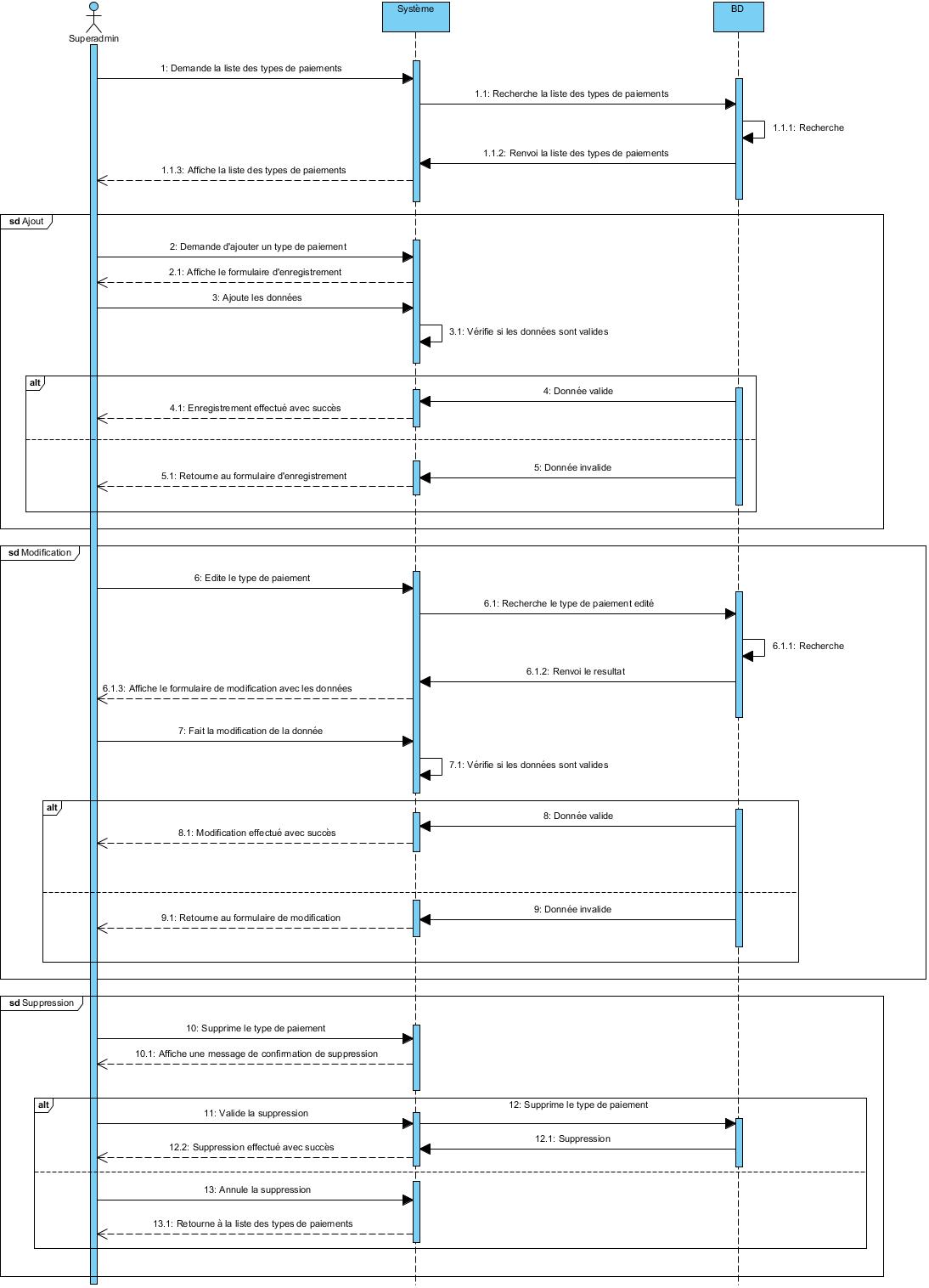


Figure 12 : Diagramme de séquence pour le cas d’utilisation « Gérer les types de paiements »

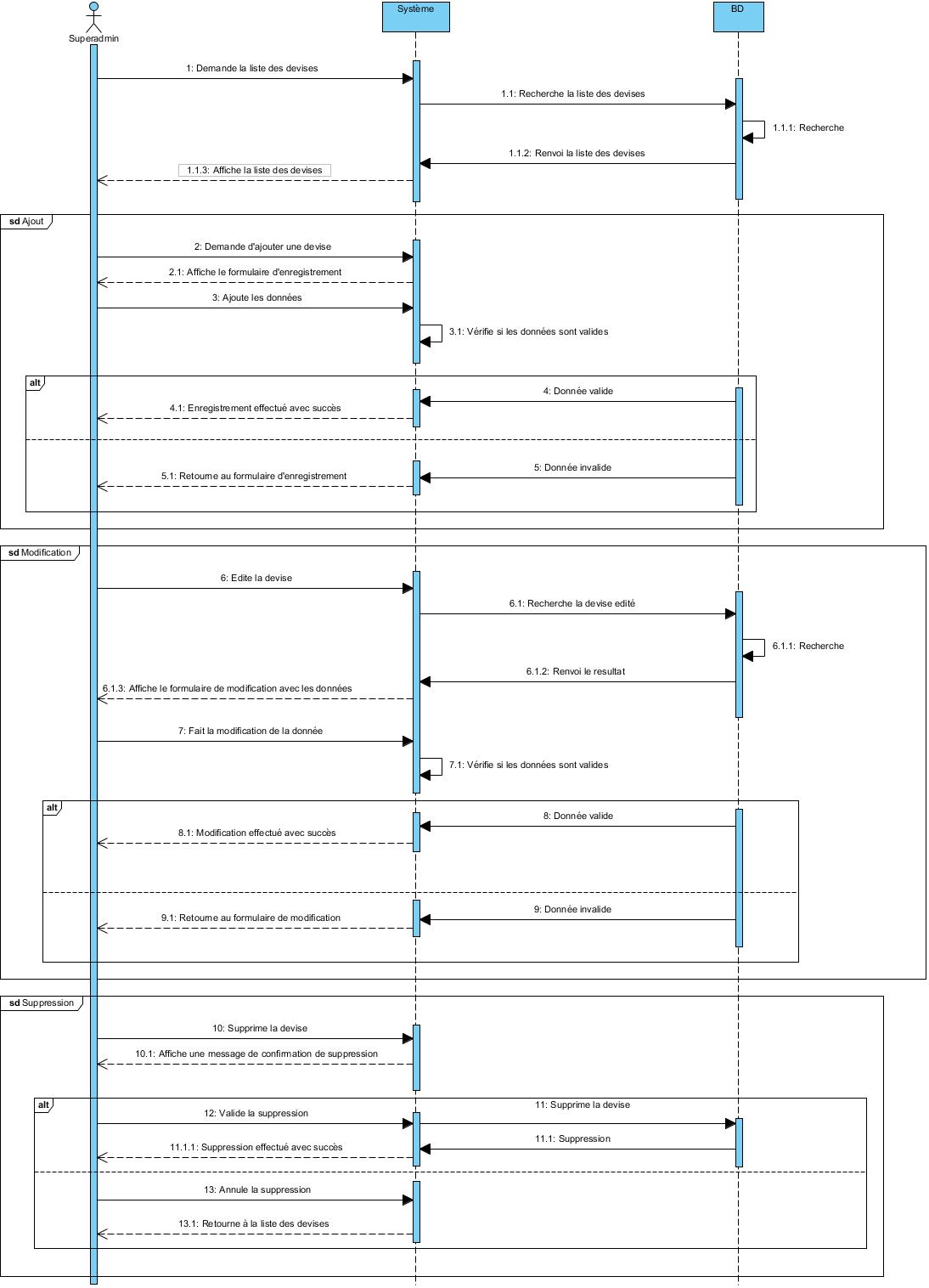
La figure 13 représente le diagramme de séquence pour le cas d’utilisation « Gérer les devises ».

Figure 13 : Diagramme de séquence pour le cas d’utilisation « Gérer les devises »

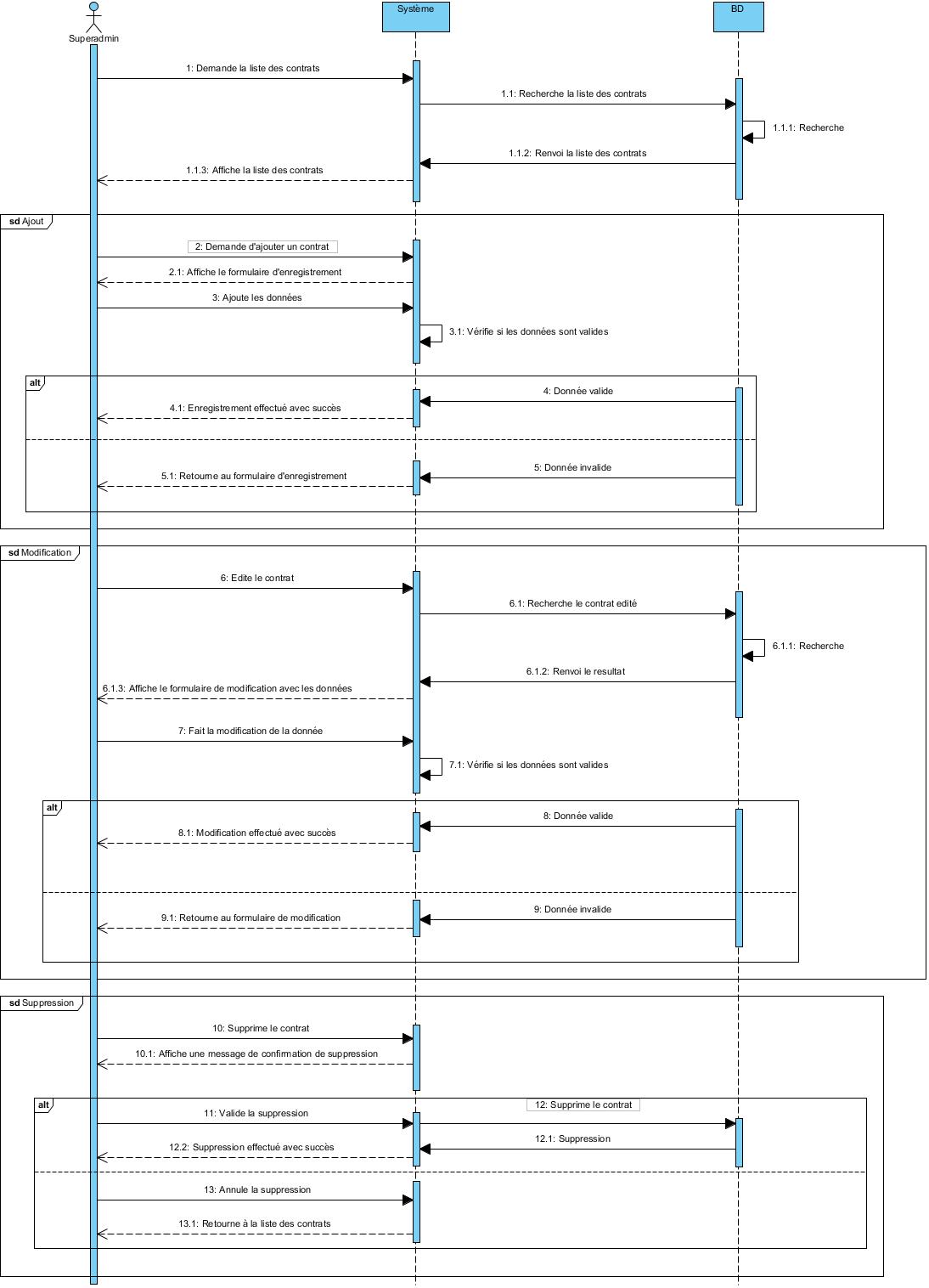
La figure 14 représente le diagramme de séquence pour le cas d’utilisation « Gérer les contrats ».

Figure 14: Diagramme de séquence pour le cas d’utilisation « Gérer les contrats »

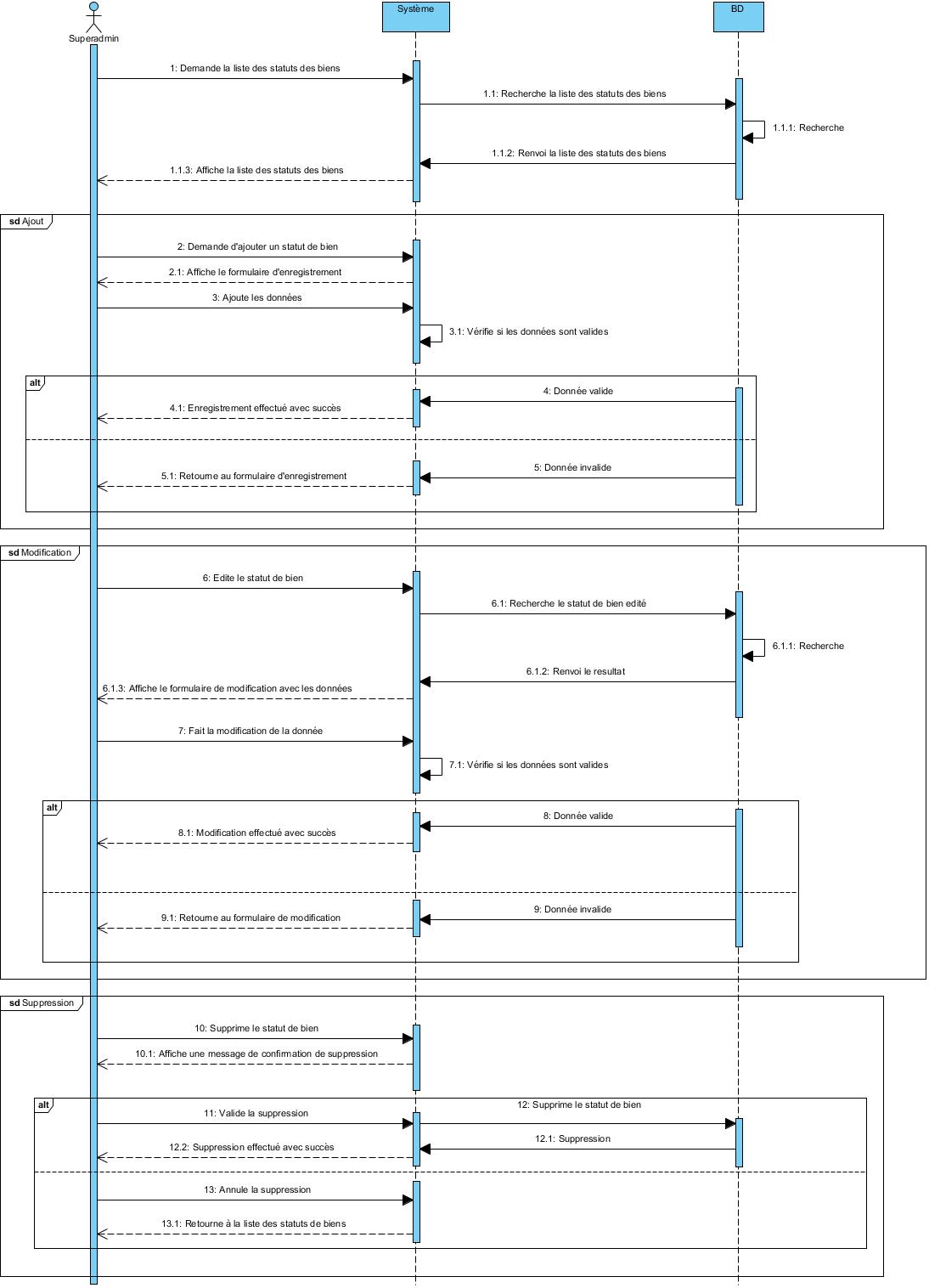
La figure 15 représente le diagramme de séquence pour le cas d’utilisation « Gérer les statuts de biens ».

Figure 15: Diagramme de séquence pour le cas d’utilisation « Gérer les statuts de biens »

La figure 16 représente le diagramme de séquence pour le cas d’utilisation « Gérer les biens ».

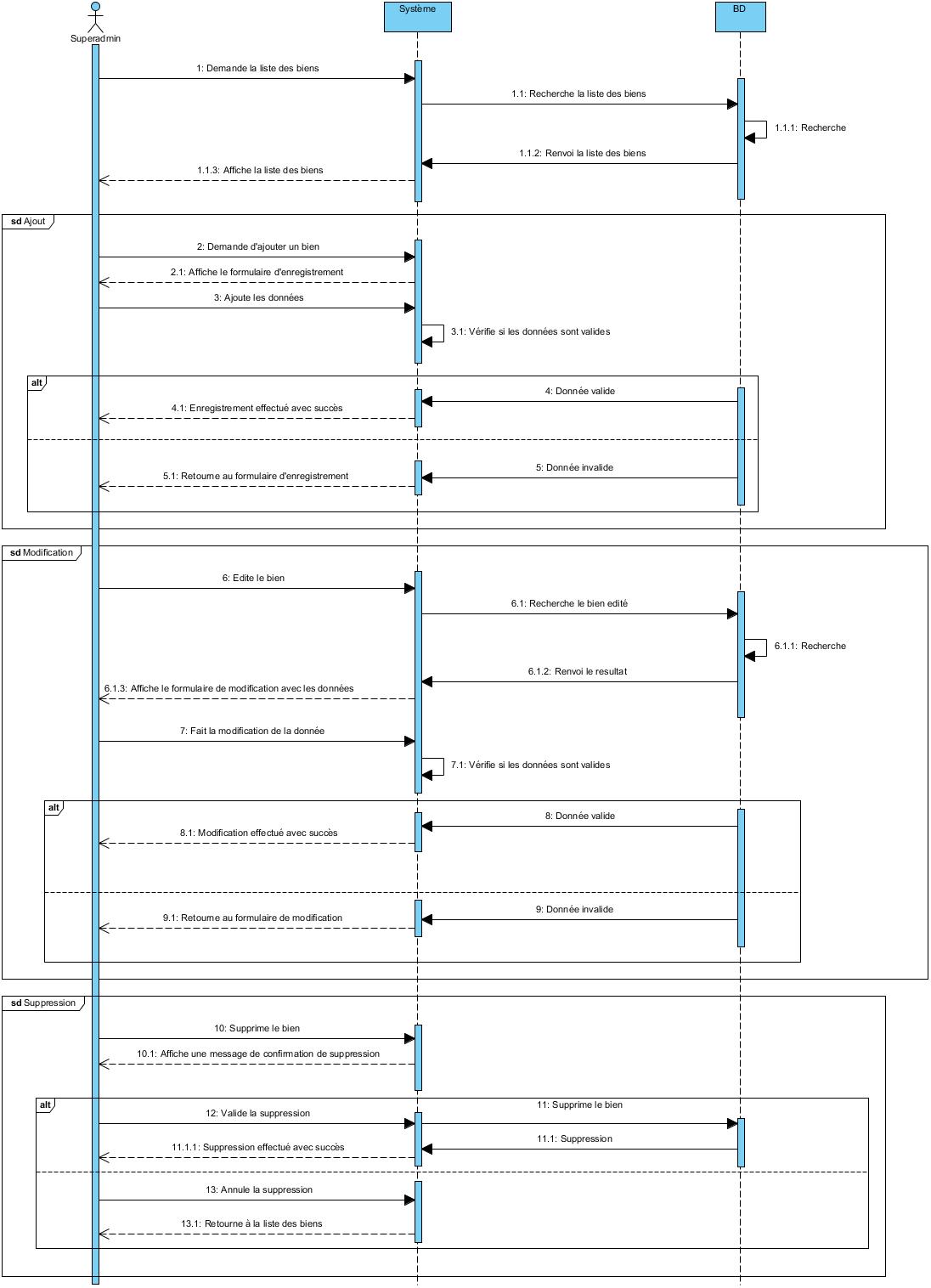


Figure 16: Diagramme de séquence pour le cas d’utilisation « Gérer les biens »

La figure 17 représente le diagramme de séquence pour le cas d’utilisation « Gérer les entités ».

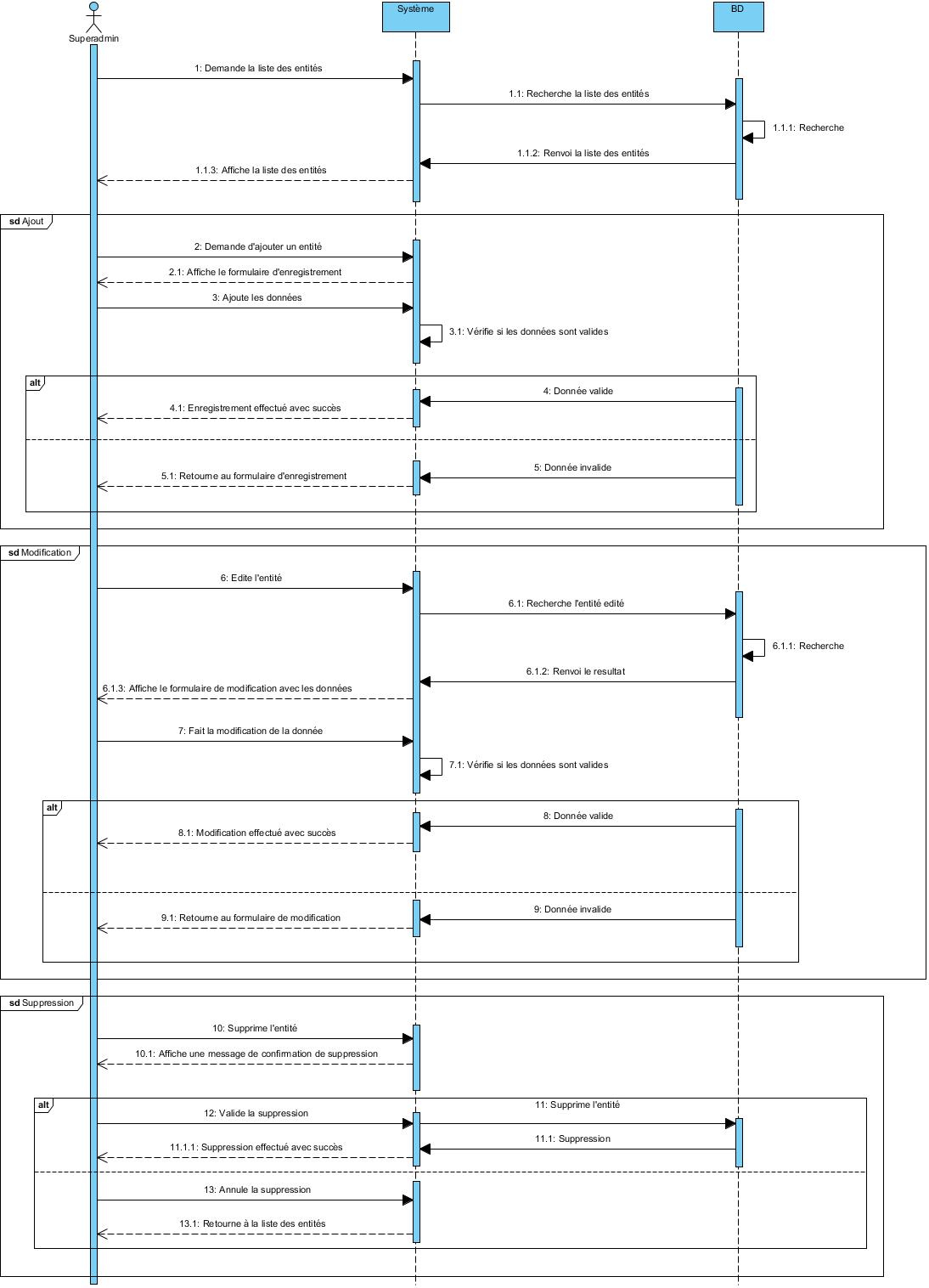


Figure 17: Diagramme de séquence pour le cas d’utilisation « Gérer les entités »

La figure 18 représente le diagramme de séquence pour le cas d’utilisation « Gérer les agences ».

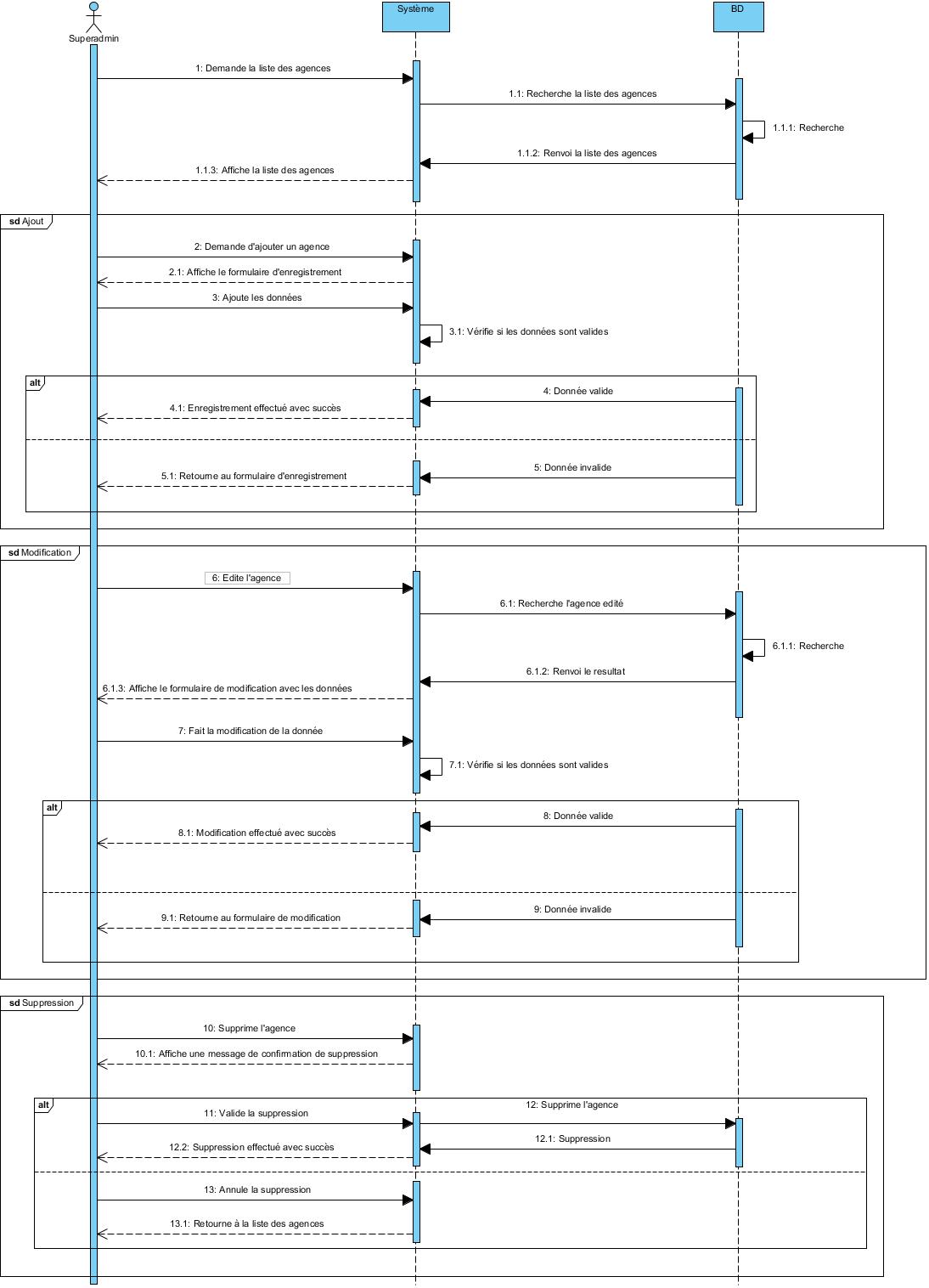


Figure 18 : Diagramme de séquence pour le cas d’utilisation « Gérer les agences »

La figure 19 représente le diagramme de séquence pour le cas d’utilisation « Gérer les terrains ».

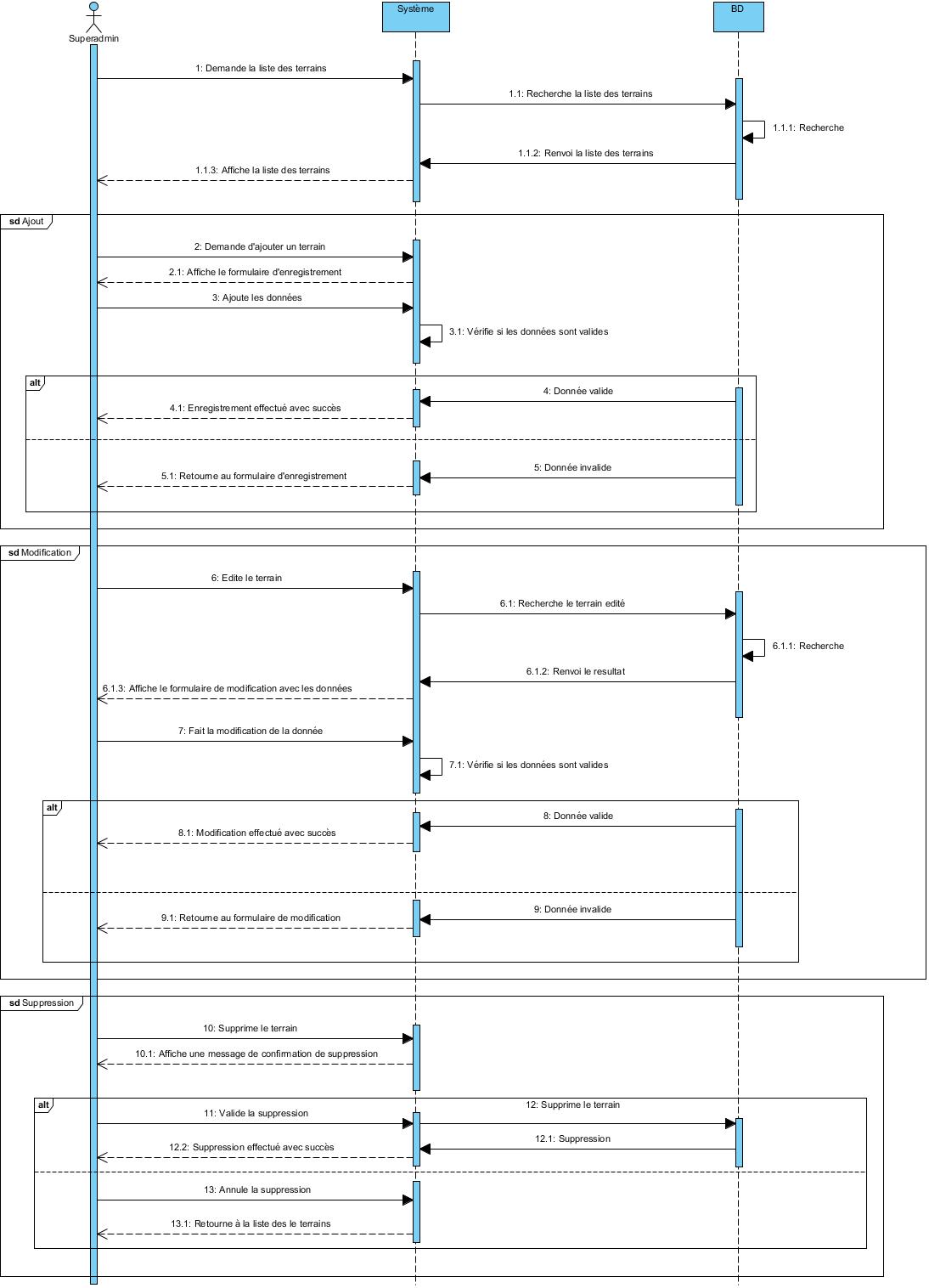


Figure 19: Diagramme de séquence pour le cas d’utilisation « Gérer les terrains »

### 6.5 Spécification des besoins techniques

#### 6.5.1 Capture de besoins techniques

Ce sont des exigences qui ne concernent pas spécifiquement le comportement du système mais plutôt ils identifient des contraintes internes et externes du système. Elles permettent d’exprimer les contraintes géographiques, organisationnelles et techniques relatives à la configuration du réseau matériel.

#### 6.5.2 Diagramme de cas d’utilisation technique :

La figure 20 représente le diagramme de cas d’utilisation technique du projet.

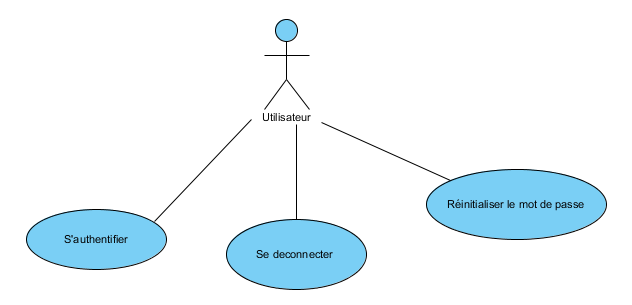


Figure 20 : Diagramme de cas d’utilisation technique

### 6.6 Modélisation du domaine

L’élaboration du modèle des classes du domaine permet d’opérer une transition vers une véritable modélisation objet. L’analyse du domaine est une étape totalement séparée de l’analyse des besoins. Elle peut être menée avant, en parallèle ou après cette dernière. La phase d’analyse du domaine permet d’élaborer la première version du diagramme de classe appelée modèle du domaine.

Ce modèle doit définir les classes qui modélisent les entités ou concepts présents dans le domaine de l’application [8]. Il s’agit donc de produire un modèle des objets du monde réel dans un domaine donné. Ces entités ou concepts sont identifiés directement à partir de la connaissance du domaine ou par des entretiens avec des experts du domaine.

Pour établir le diagramme, on suit des étapes :

* Identifier les entités ou concepts du domaine.
* Identifier et ajouter les associations et les attributs.
* Organiser et simplifier le modèle en éliminant les classes redondantes et en utilisant l’héritage.
* Le cas échéant, structurer les classes en paquetage selon les principes de cohérence et d’indépendance.

La figure 21 illustre le modèle de domaine du système.

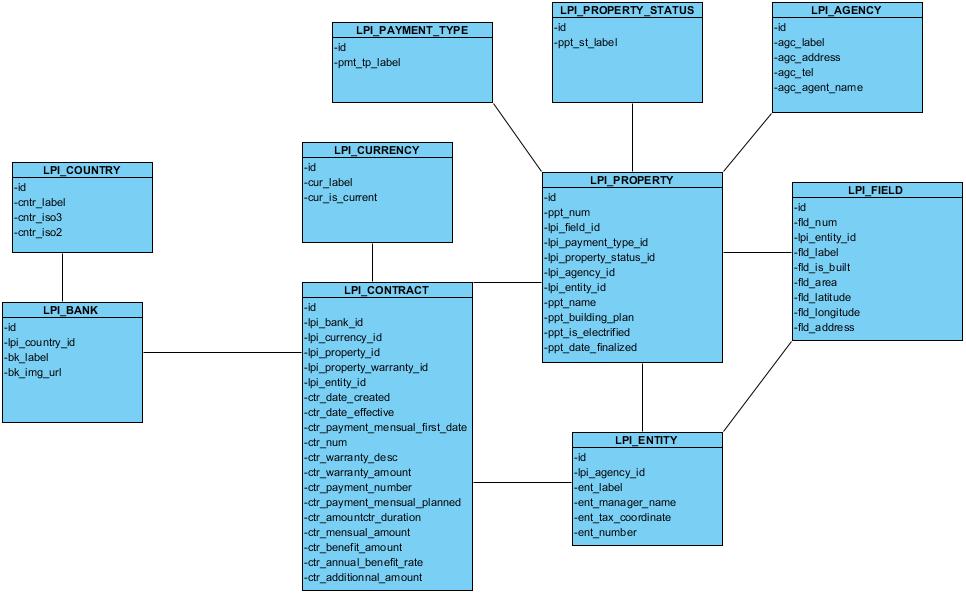


Figure 21: Modèle de domaine du système

## Chapitre 7 : Conception détaillée

### 7.1 Architecture du Système

L’architecture du système se base sur le pattern MVC (Modèle-Vue-Contrôleur). Le patron Modèle-Vue-Contrôleur (Model-View-Controller), tout comme les patrons modèle vue- présentation ou Présentation, abstraction, contrôle, est un modèle destiné à répondre aux besoins des applications interactives en séparant les problématiques liées aux différents composants au sein de leur architecture respective. [3] MVC est un patron de conception permettant de séparer le code d’accès aux données, la présentation et le contrôle de l’ensemble des actions. Ce paradigme regroupe les fonctions nécessaires en trois catégories :

* Le Modèle (modèle de données).
* La Vue (présentation, interface utilisateur).
* Le Contrôleur (logique de contrôle, gestion des événements, synchronisation).

Le mécanisme de l’architecture MVC est représenté dans la figure 22.

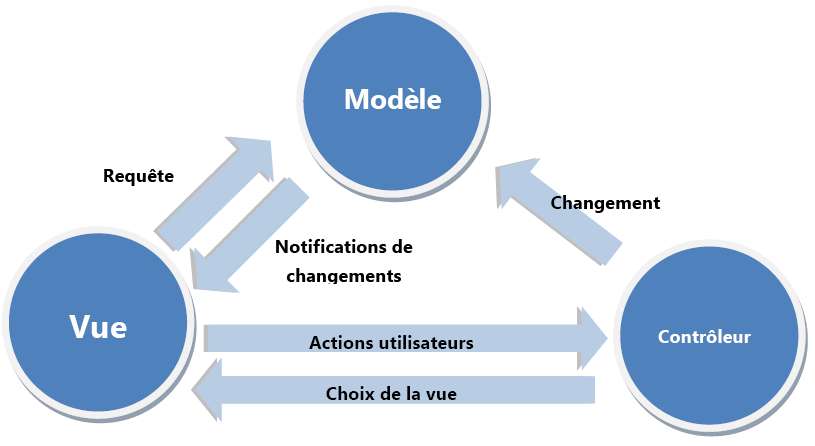


Figure 22 : Mécanisme de l’architecture MVC

* **Le contrôleur**

Le contrôleur est l’organe de contrôle du système. Il a en charge l’analyse des demandes clientes et l’orchestration des appels aux vues et modèles nécessaires à la fourniture de la réponse attendue. Il doit aussi faire le choix des appels et l’ordre de ceci.

* **Le modèle**

Le modèle est l’organe de récupération et de sélection des données pertinentes pour répondre à la demande. Son rôle consiste à récupérer, filtrer, modifier les données afin de fournir un sous ensemble de données pertinentes pour la réponse.

* **La vue**

La vue est l’organe en charge de produire la présentation des résultats en fonction de données qui lui sont fournies. Pour cela la vue s’appuie généralement sur des « Template » ou modèle de réponse auquel elle intègre les données afin de composer le résultat final.

**Avantage :**

Un avantage apporté par ce modèle est la clarté de l'architecture qu'il impose. Cela simplifie la tâche du développeur qui tenterait d'effectuer une maintenance ou une amélioration sur le projet.

En effet, la modification des traitements ne change en rien la vue. Par exemple on peut passer d'une base de données de type SQL à XML en changeant simplement les traitements d'interaction avec la base, et les vues ne s'en trouvent pas affectées. L’avantage du MVC est que celui-ci permet plus de souplesse et de maintenabilité dans la durée de vie d’une application [3].

### 7.2 Diagramme de séquence de conception pour chaque cas d'utilisation

Le diagramme de séquence de conception est un diagramme de séquence qui est plus détaillé. En effet, le diagramme de séquence système ne montre que les scénarios d’échange de messages entre les acteurs et le système tandis que le diagramme de séquence de conception permet de donner une vue « en largeur » du déroulement d’une opération en montrant tous les objets impliqués et en donnant plus de détails sur la façon dont cette méthode procède.

La figure 23 illustre le diagramme de séquence de conception « Gérer les banques ».

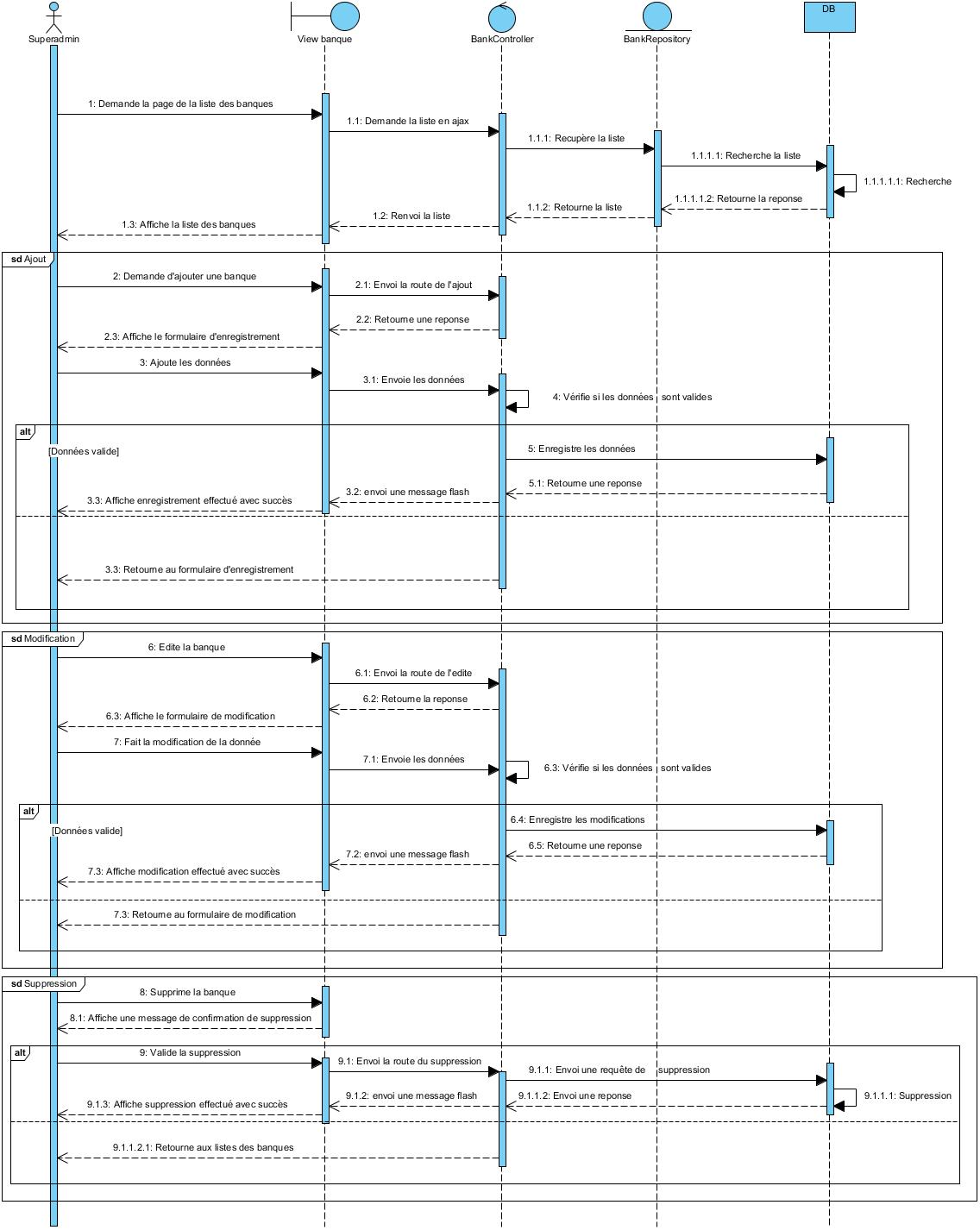


Figure 23: Diagramme de séquence de conception « Gérer les banques »

La figure 24 illustre le diagramme de séquence de conception « Gérer les types de paiements ».

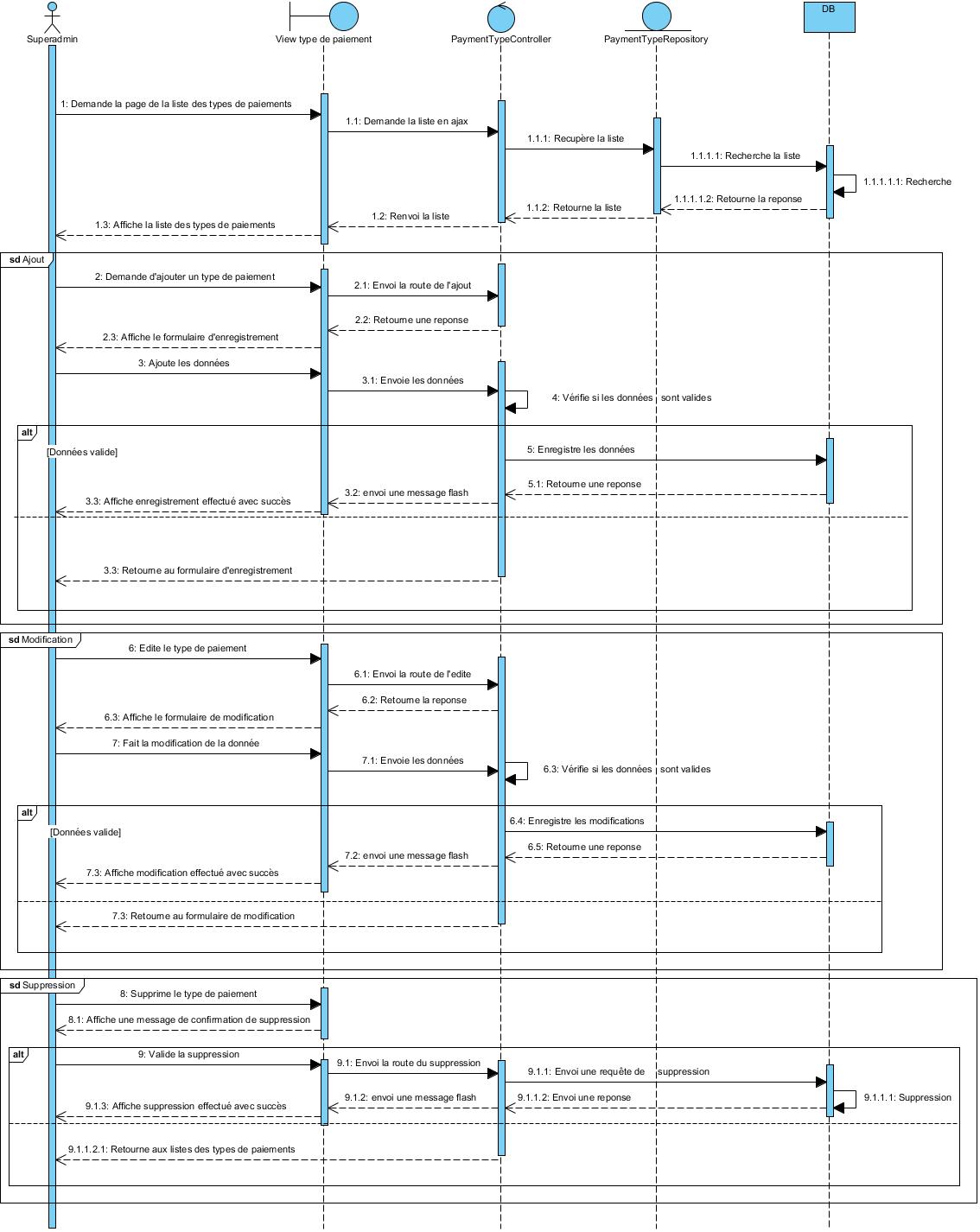


Figure 24 : Diagramme de séquence de conception « Gérer les types de paiements »

La figure 25 illustre le diagramme de séquence de conception « Gérer les devises ».

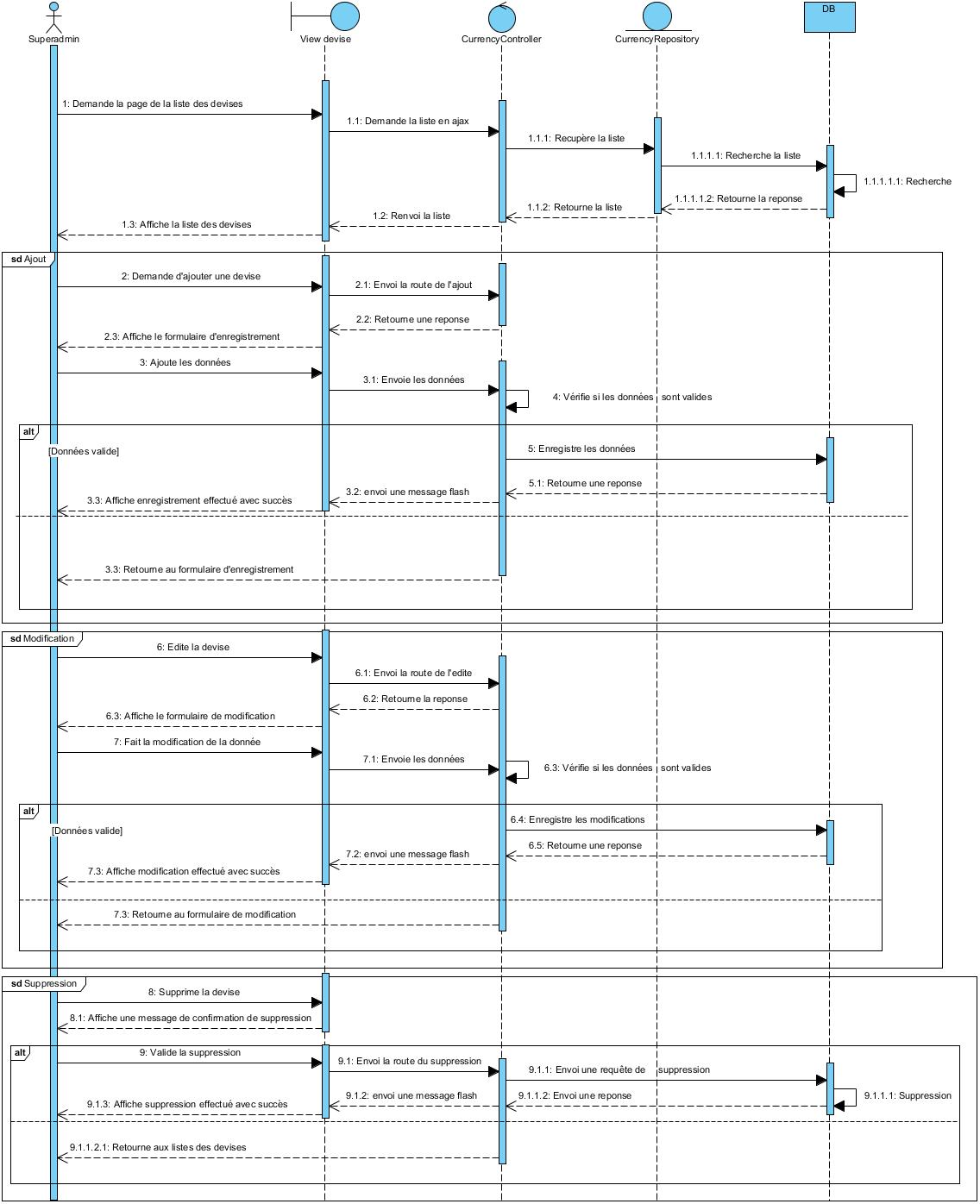


Figure 25 : Diagramme de séquence de conception « Gérer les devises »

La figure 26 illustre le diagramme de séquence de conception « Gérer les contrats ».

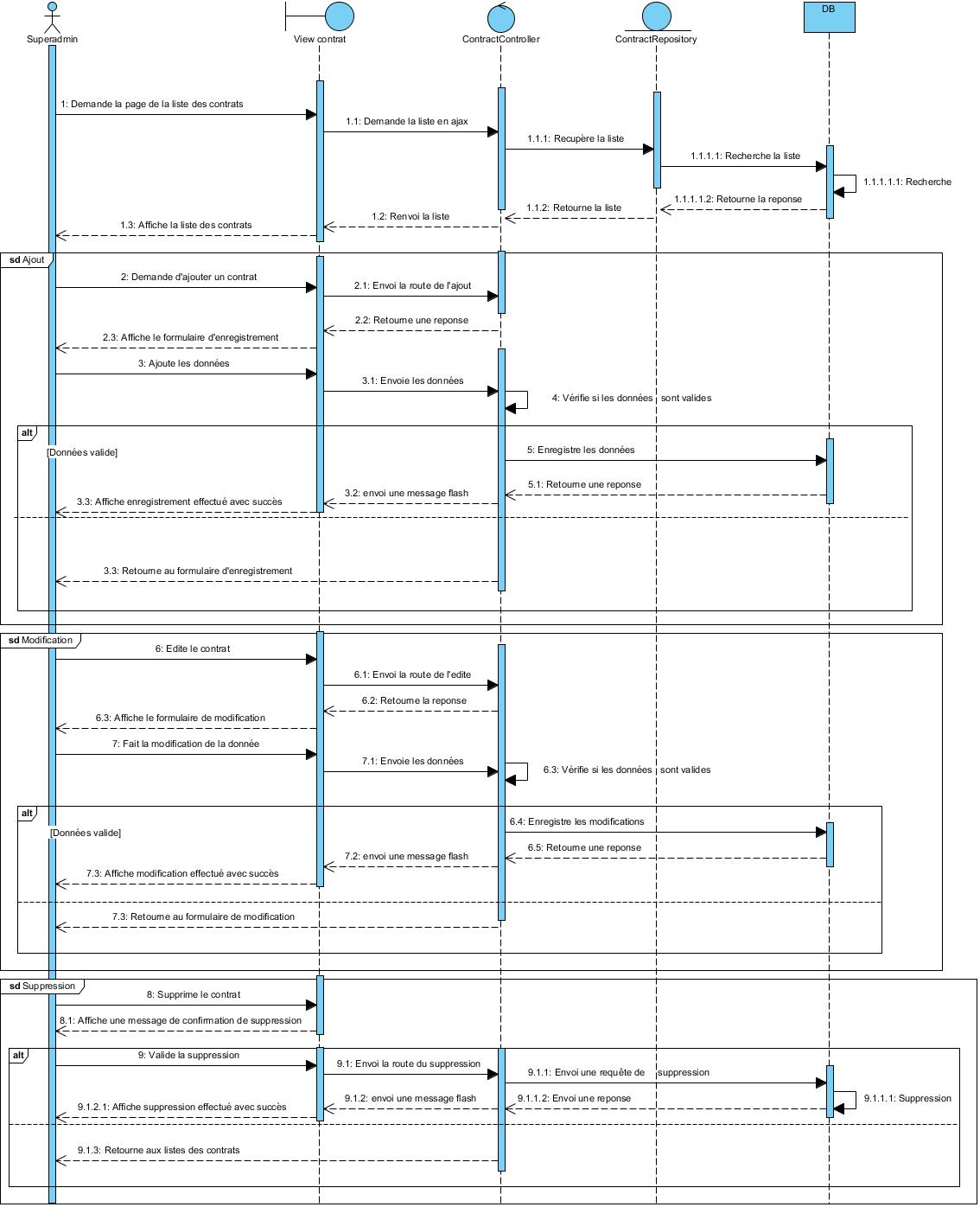


Figure 26 : Diagramme de séquence de conception « Gérer les contrats »

La figure 27 illustre le diagramme de séquence de conception « Gérer les statuts de biens ».

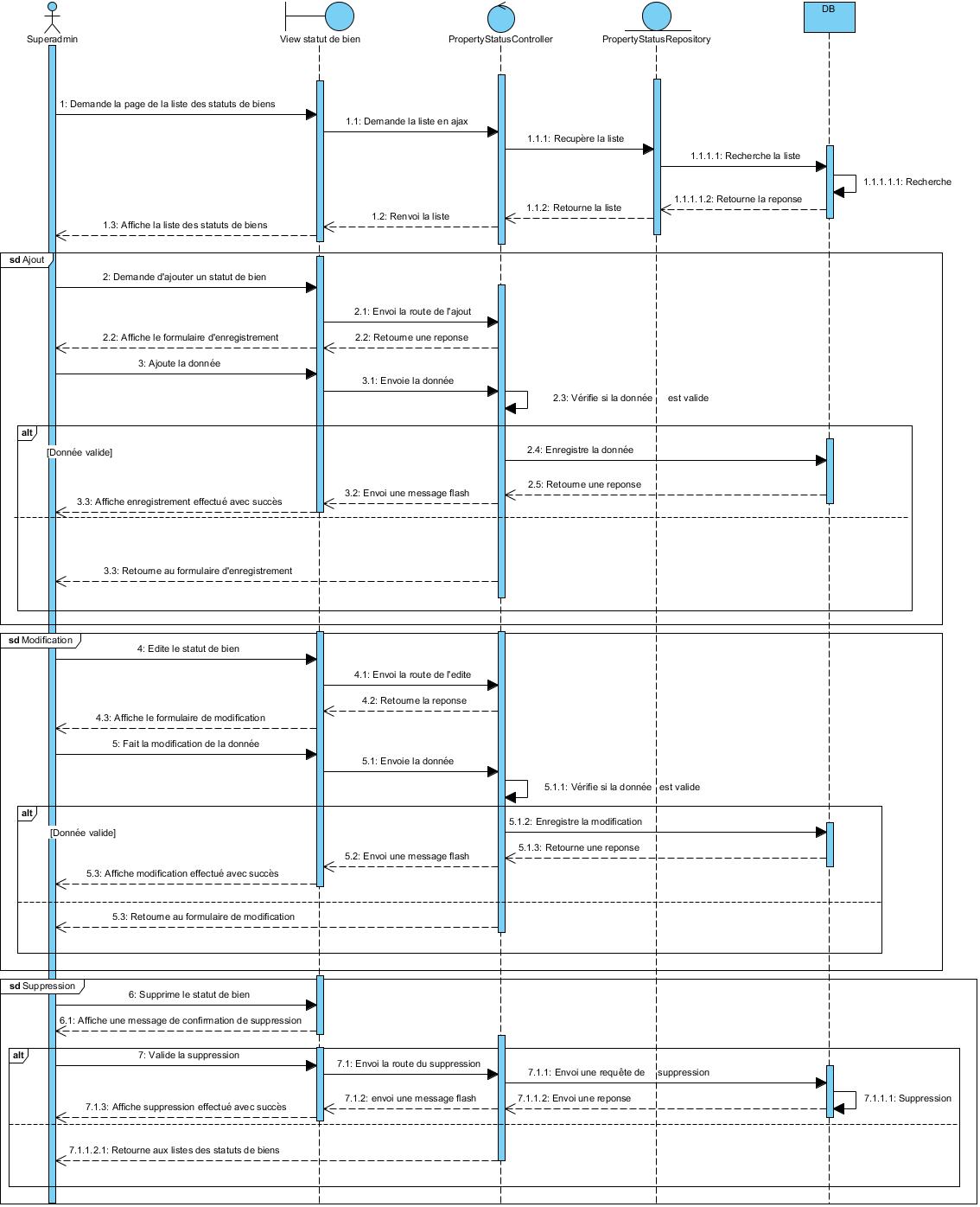


Figure 27 : Diagramme de séquence de conception « Gérer les statuts de biens »

La figure 28 illustre le diagramme de séquence de conception « Gérer les biens ».

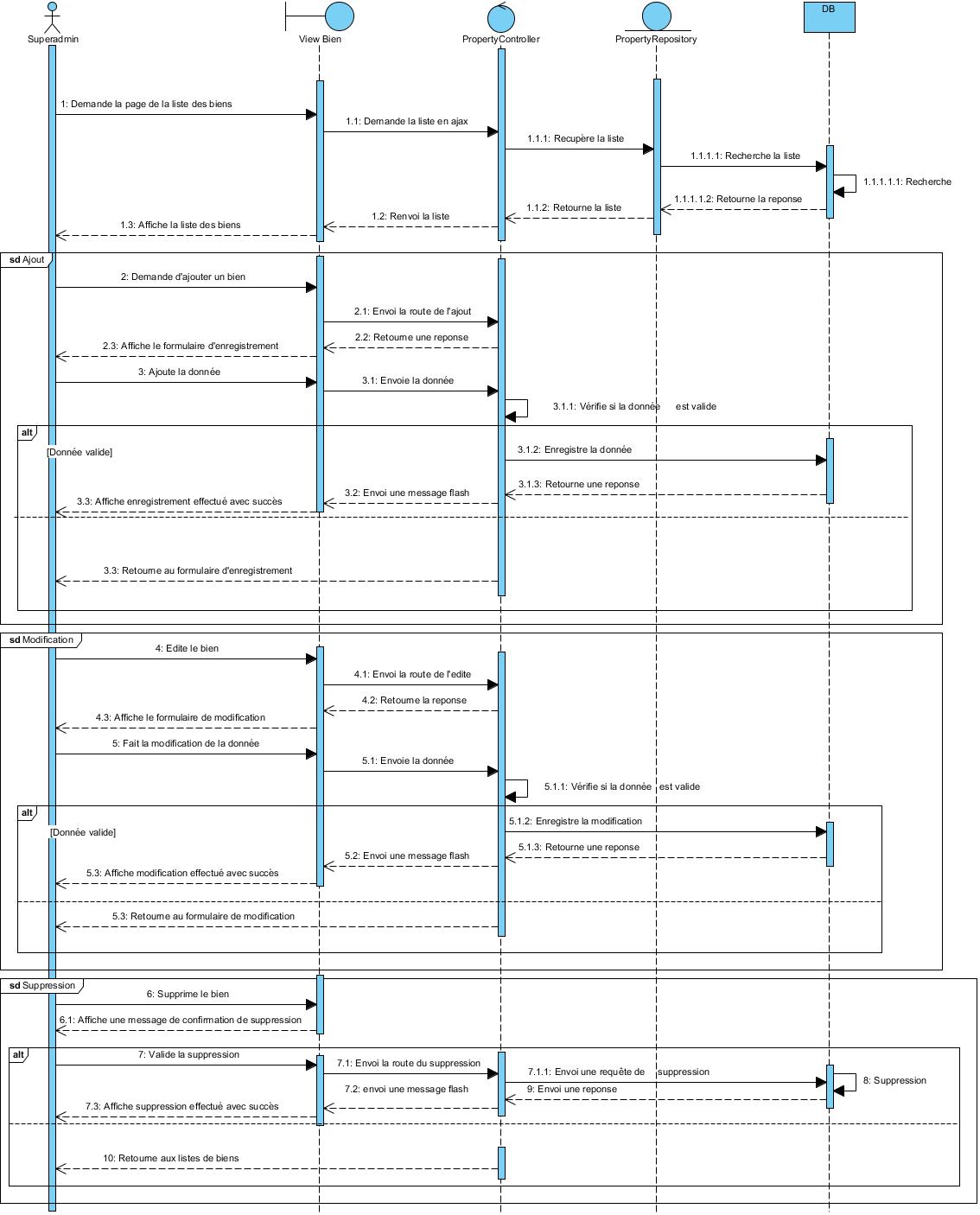


Figure 28 : Diagramme de séquence de conception « Gérer les biens »

La figure 29 illustre le diagramme de séquence de conception « Gérer les agences ».

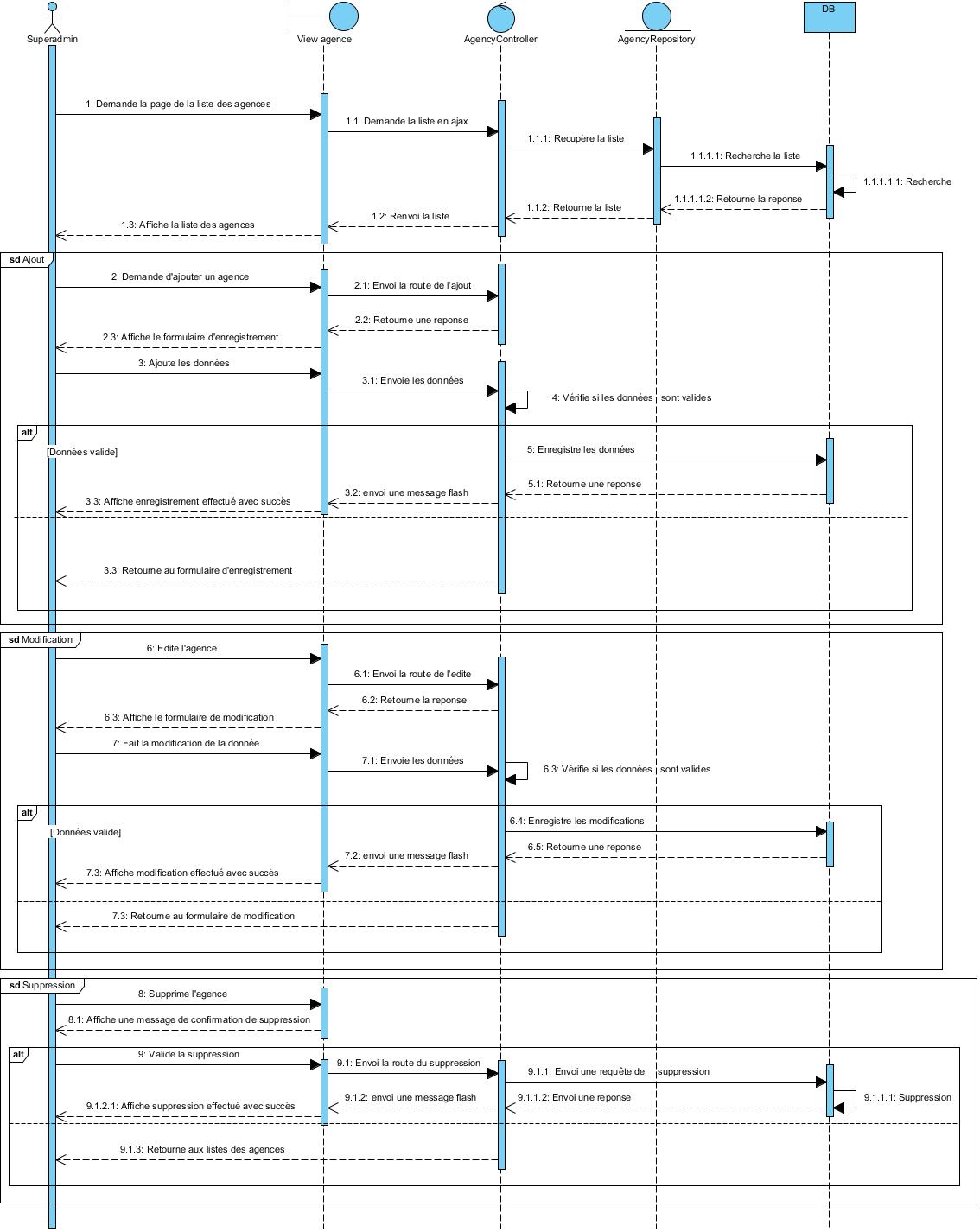


Figure 29 : Diagramme de séquence de conception « Gérer les agences »

La figure 30 illustre le diagramme de séquence de conception « Gérer les entités ».

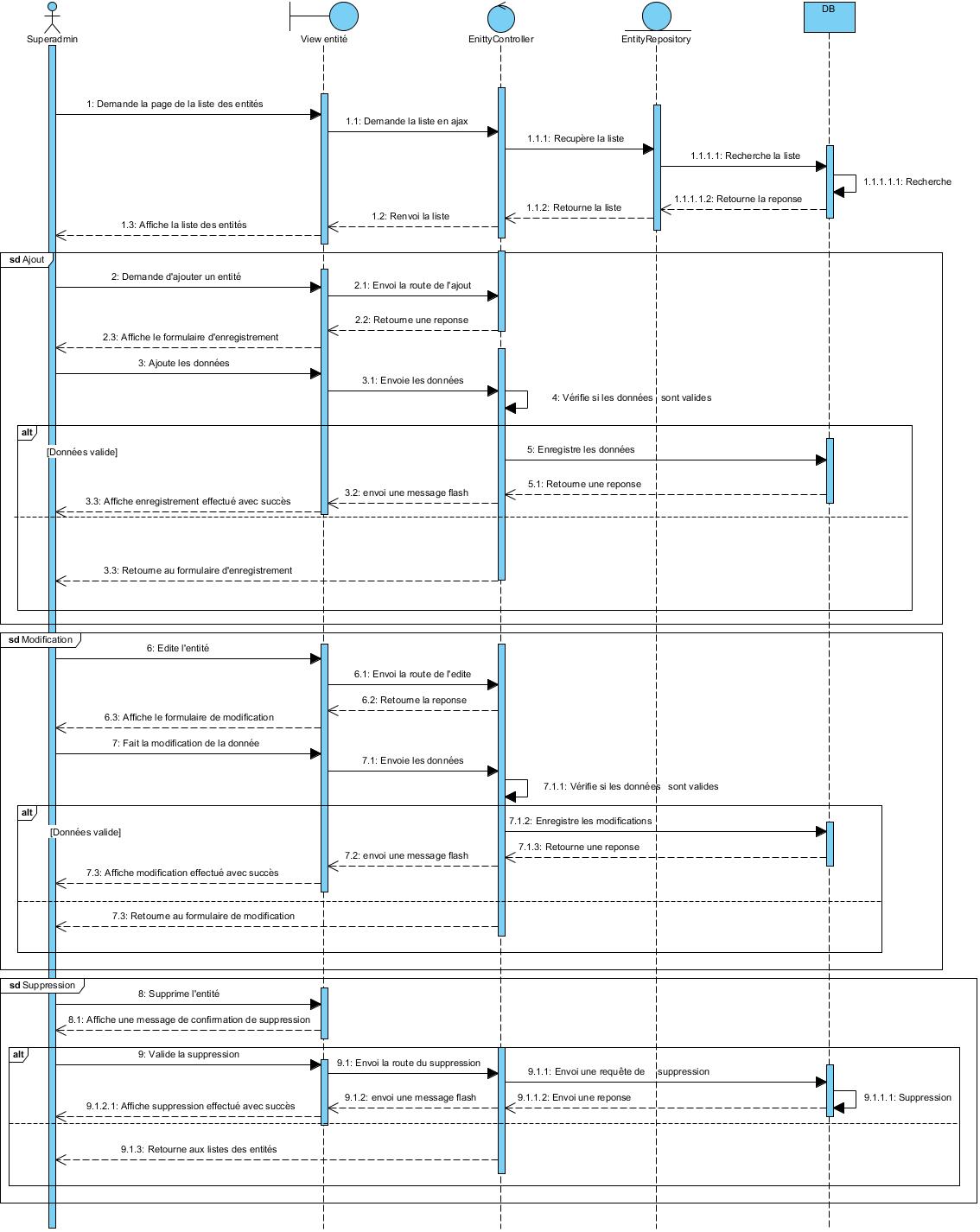


Figure 30 : Diagramme de séquence de conception « Gérer les entités »

La figure 31 illustre le diagramme de séquence de conception « Gérer les terrains ».

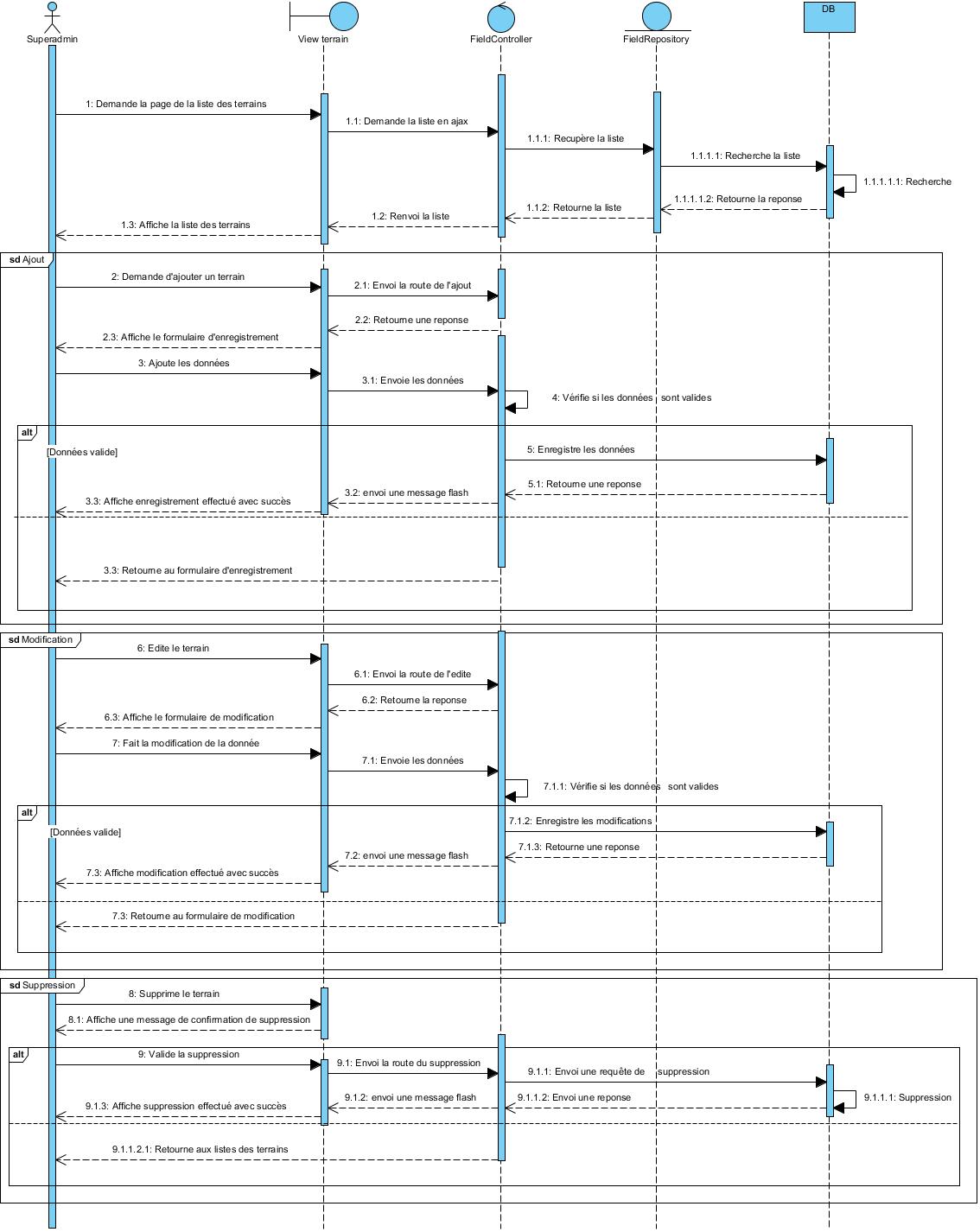


Figure 31 : Diagramme de séquence de conception « Gérer les terrains »

### 7.3 Diagramme de classe de conception pour chaque cas d'utilisation.

La figure 32 représente le diagramme de classe de conception de « Gérer les banques ».



Figure 32 : Diagramme de classe de conception de « Gérer les banques »

La figure 33 représente le diagramme de classe de conception de « Gérer les entités».



Figure 33 : Diagramme de classe de conception de « Gérer les entités»

La figure 34 représente le diagramme de classe de conception de « Gérer les terrains ».



Figure 34 : Diagramme de classe de conception de « Gérer les terrains »

La figure 35 représente le diagramme de classe de conception de « Gérer les biens ».

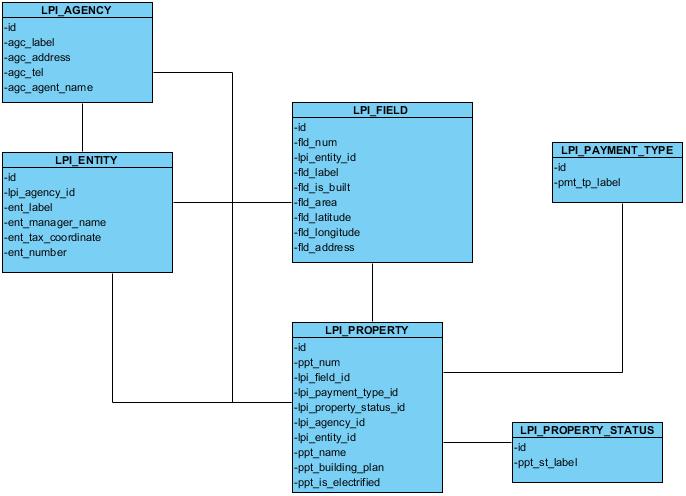


Figure 35 : Diagramme de classe de conception de « Gérer les biens »

La figure 36 représente le diagramme de classe de conception de « Gérer les contrats».

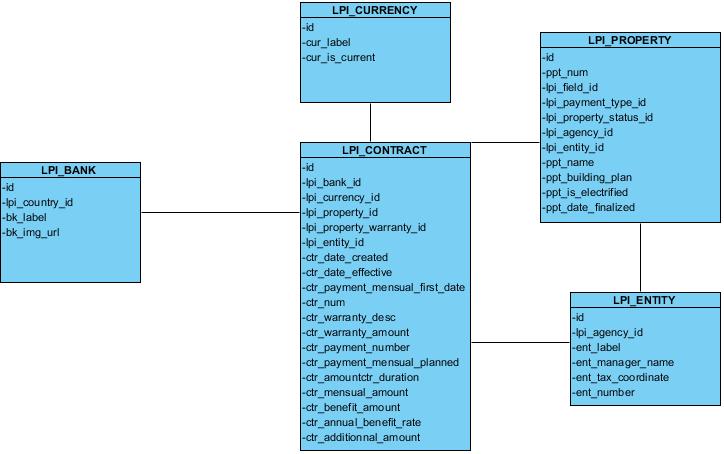


Figure 36 : Diagramme de classe de conception de « Gérer les contrats »

### 7.4 Diagramme de classe de conception

La figure 37 représente le diagramme de classe de conception de l’application.

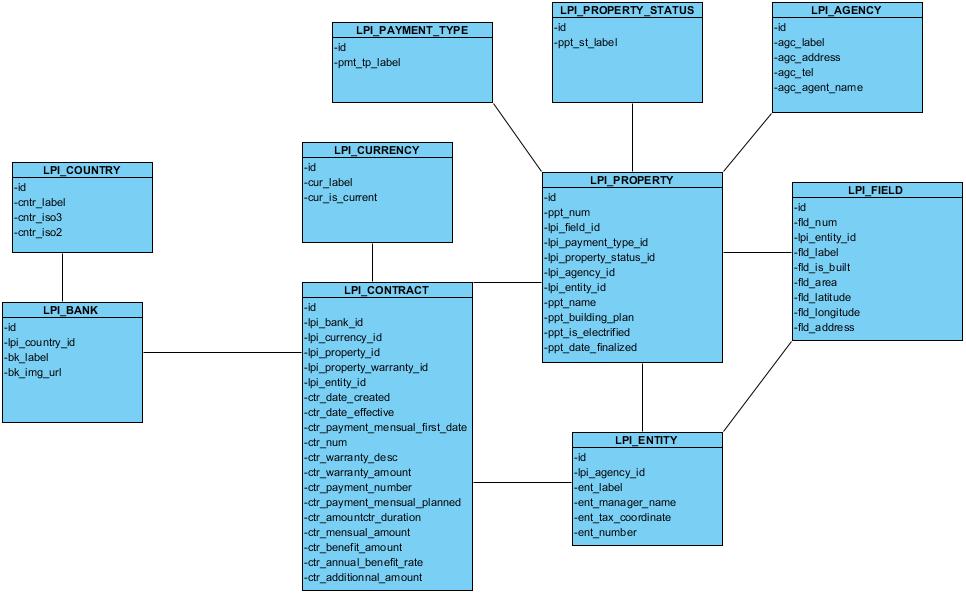


Figure 37 : Diagramme de classe de conception de l’application

### 7.5 Diagramme de paquetage

Un paquetage regroupe des éléments de la modélisation appelés aussi membres, portant sur un sous-ensemble du système. Le découpage en paquetage doit traduire un découpage logique du système à construire qui corresponde à des espaces de nommage homogènes.

Un paquetage peut importer des éléments d’un autre paquetage. Un paquetage peut être fusionné avec un autre paquetage [1].

La figure 38 représente le diagramme de paquetage de l’application.



Figure 38 : Diagramme de paquetage de l’application

### 7.6 Diagramme de déploiement

Le diagramme de déploiement permet de représenter l’architecture physique supportant l’exploitation du système. Cette architecture comprend des nœuds correspondant aux supports physiques (serveurs, routeurs...) ainsi que la répartition des artefacts logiciels (bibliothèques, exécutables...) sur ces nœuds. C’est un véritable réseau constitué de nœuds et de connexions entre ces nœuds qui modélise cette architecture [1].

La figure 39 représente le diagramme de déploiement de l’application.

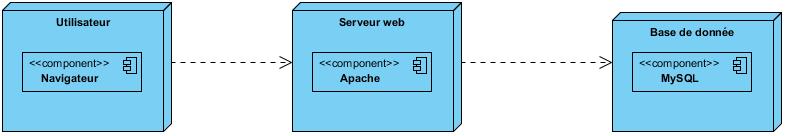


Figure 39 : Diagramme de déploiement de l’application

PARTIE III : REALISATION

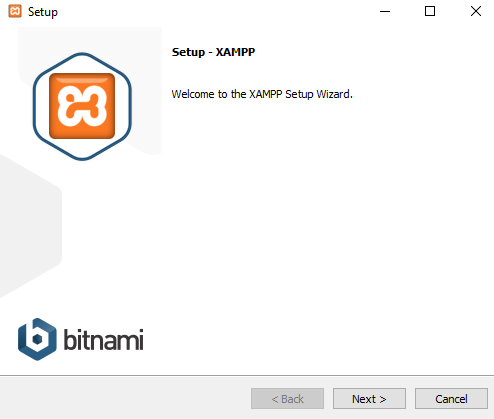
Chapitre 8 : Mise en place de l’environnement de développement

8.1. Installation et configuration des outils

8.1.1. Installation de Xampp

XAMPP est un de logiciel servant à mettre en place aisément un serveur Web, un serveur FTP et un serveur de messagerie électronique. Il possède une interface utilisateur PHPMyAdmin permettant de gérer facilement les données.

La figure 40 illustre l’installation de Xampp



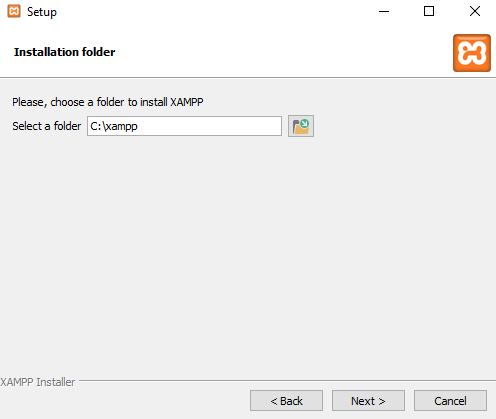


Figure 40 : Installation de Xampp

8.1.2. Installation de PHPStorm

PHPStorm est un environnement de développement intelligent, ultra-rapide et idéal pour travailler avec Symfony.

La figure 41 présente l’installation du PHPStorm.



Figure 41 : Installation du PHPStorm

8.1.3. Installation du Visual Paradigm

Visual Paradigm est un outil de modélisation pour concevoir une application.

La figure 42 presente l’installation du Visual Paradigm

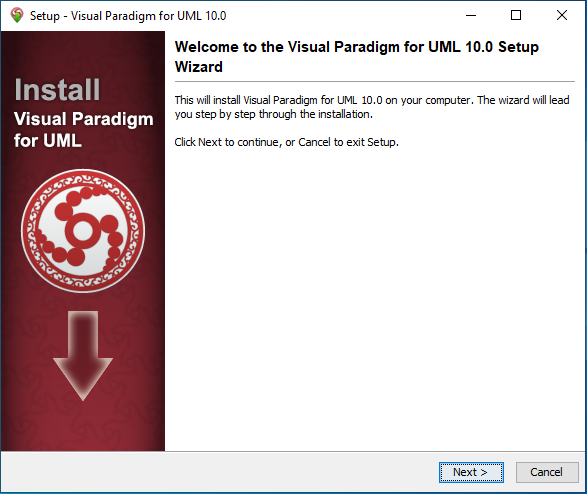


Figure 42 : Iinstallation du Visual Paradigm

8.2. Développement de l’application

8.2.1. Création de la base de données

Pour créer une base de données, il faudrait configurer le fichier .env

La figure 43 montre la configuration du fichier .env

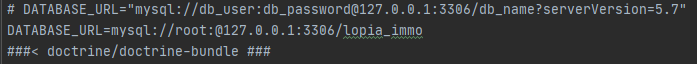


Figure 43: Configuration du fichier .env

db\_user : L’identifiant de l’utilisateur a la base de donnée

db\_password : Mot de passe de l’utilisateur

db\_name : Nom de la base de donnée

Apres cette configuration, on exécute cette commande pour créer notre base de données.

La figure 44 illustre la commande à exécuter pour créer une base de données.



Figure 44: Commande à exécuter pour créer une base de données

Apres l’exécution de cette commande, la base de données est crée

La figure 45 illustre la base de données crée



Figure 45: Illustration de la base de données crée

8.2.3. Codage de la plateforme

La figure 46 montre les constantes utilisées pour récupérer les entités.

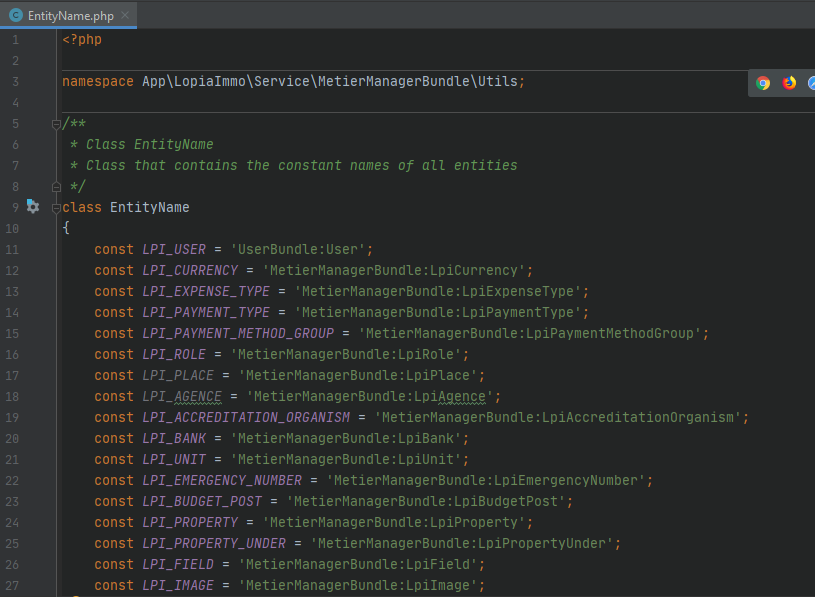


Figure 46: Constantes utilisées pour récupérer les entités

La figure 47 montre les routes de la plateforme.

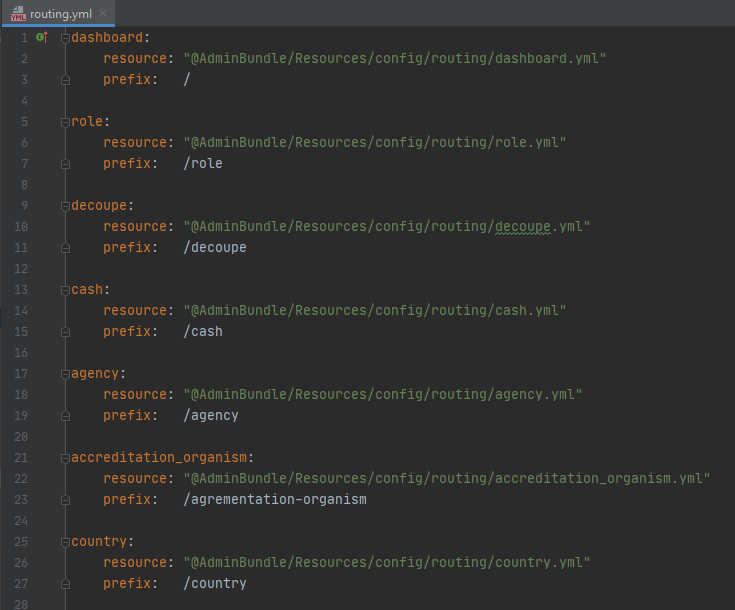


Figure 47: Les routes de la plateforme

La figure 48 illustre les routes du contrat

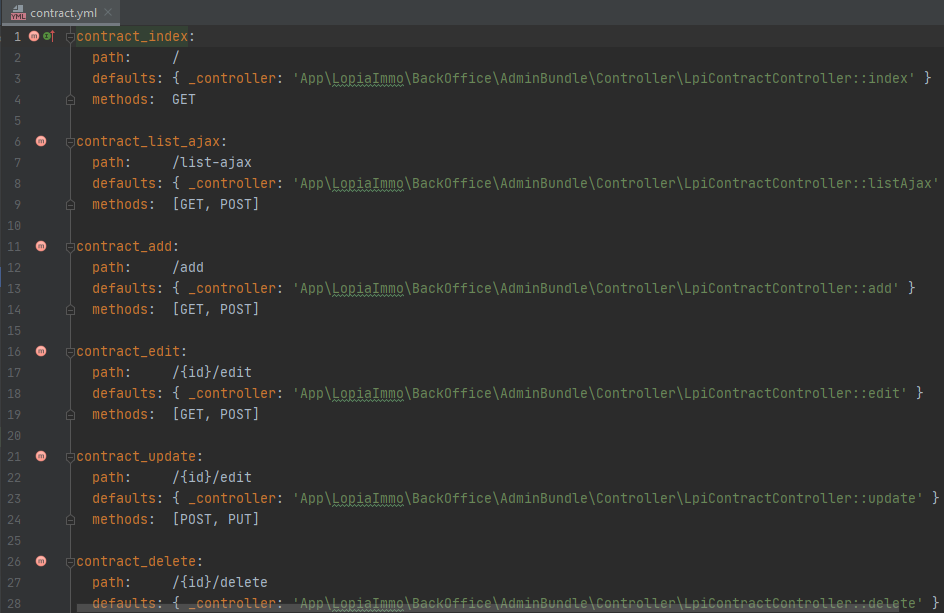
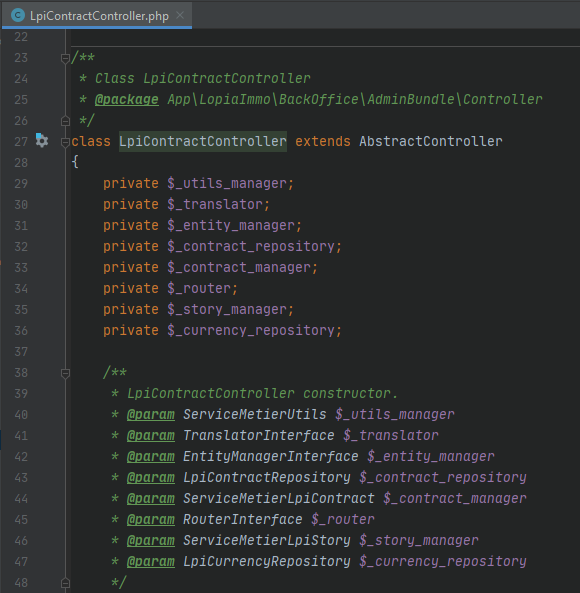
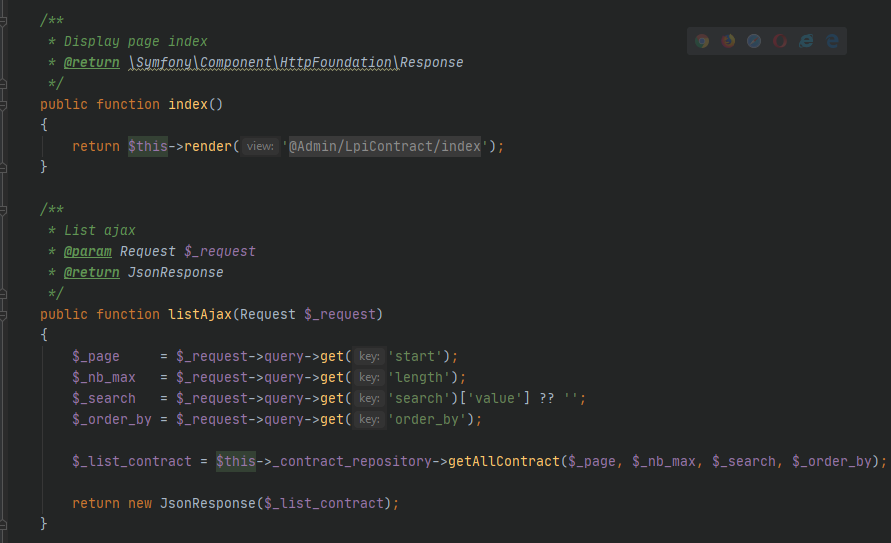


Figure 48 : Les routes du contrat

La figure 49 montre le contrôleur du contrat





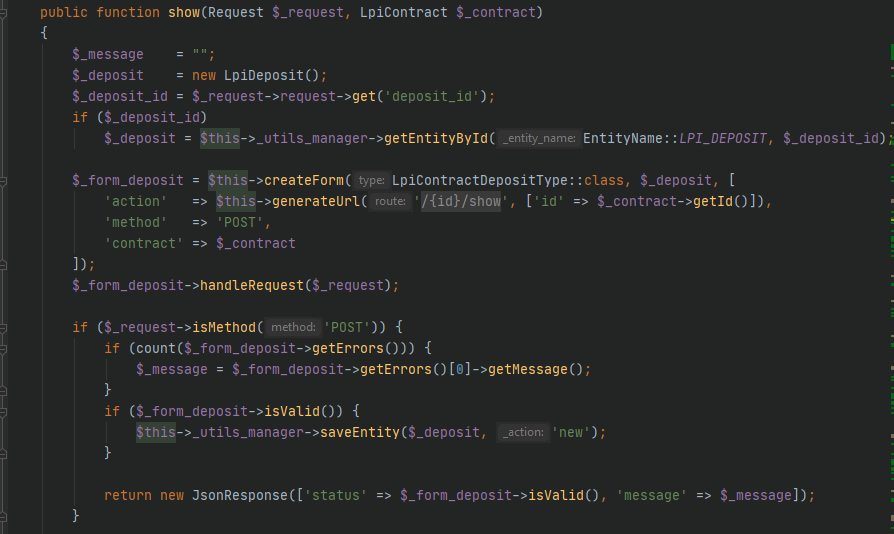
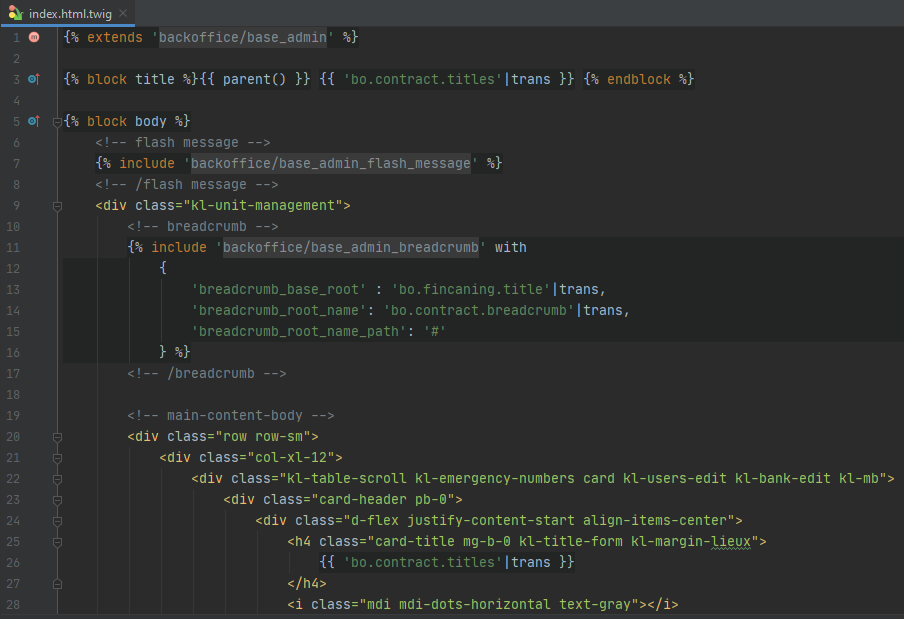
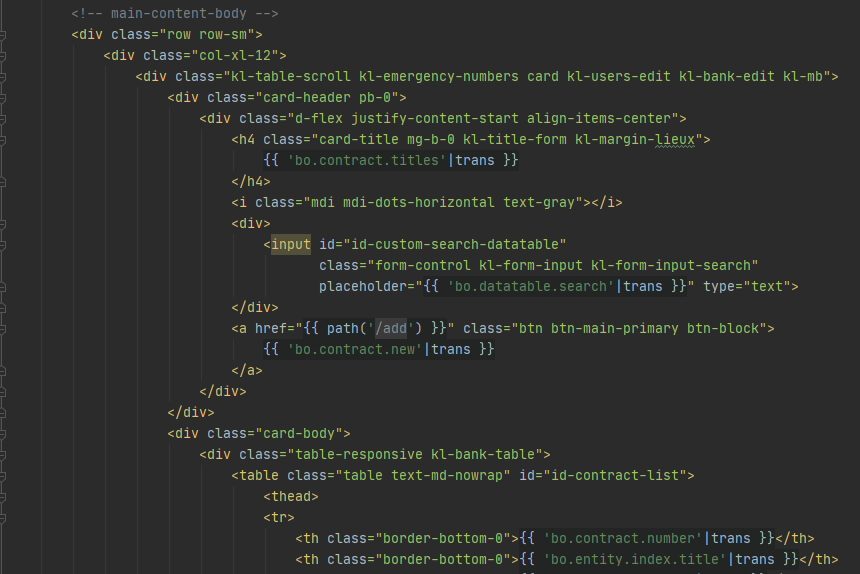


Figure 49 : Contrôleur du contrat

La figure 50 montre la vue du contrat.





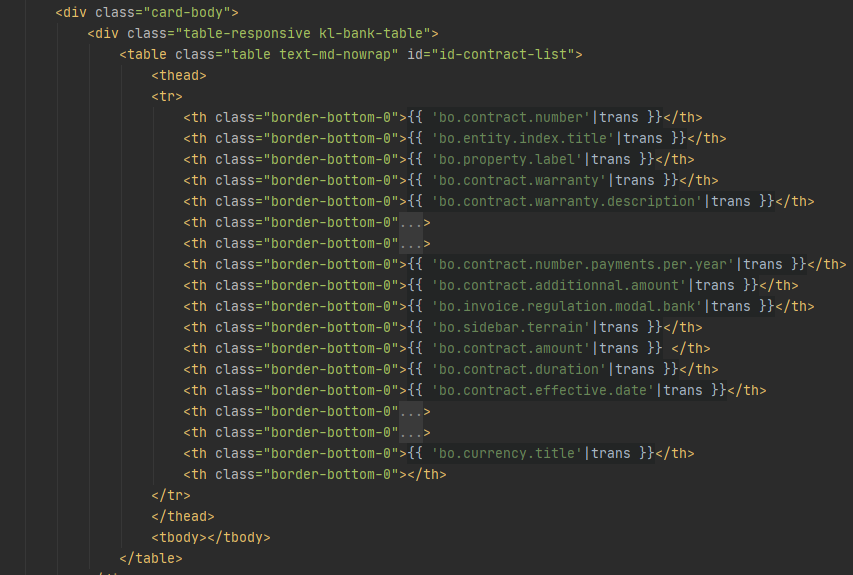
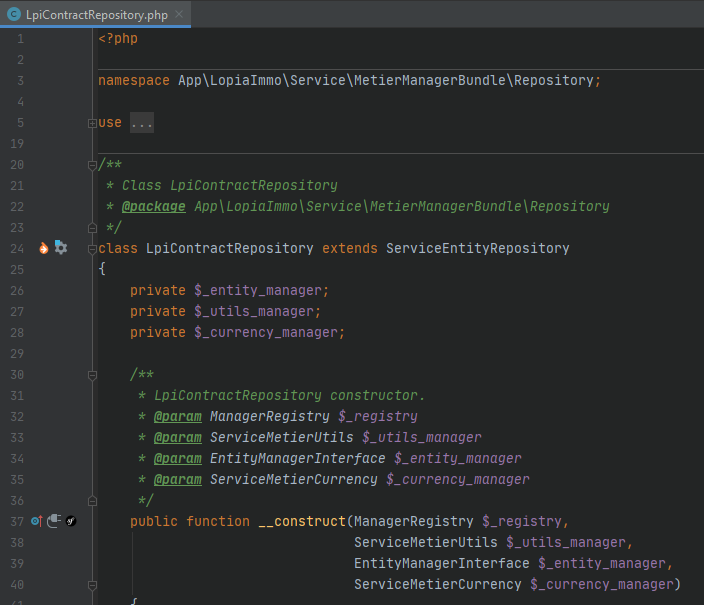
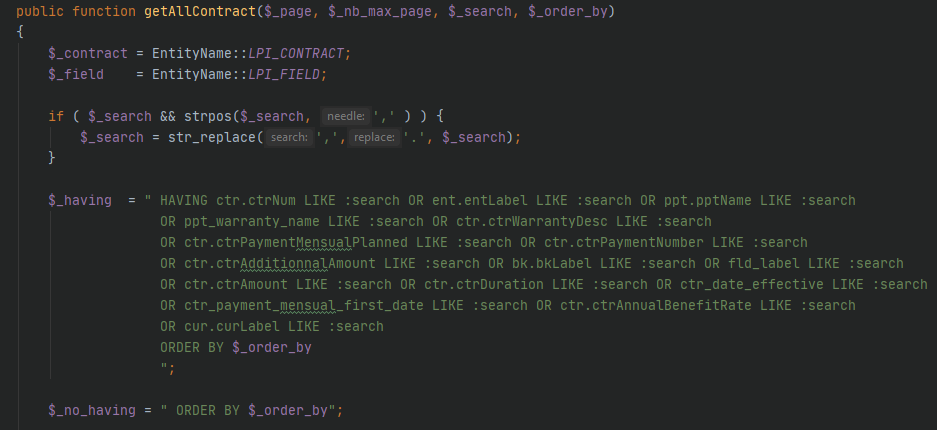




Figure 50 : Vue du contrat

La figure 51 montre la repository du contrat





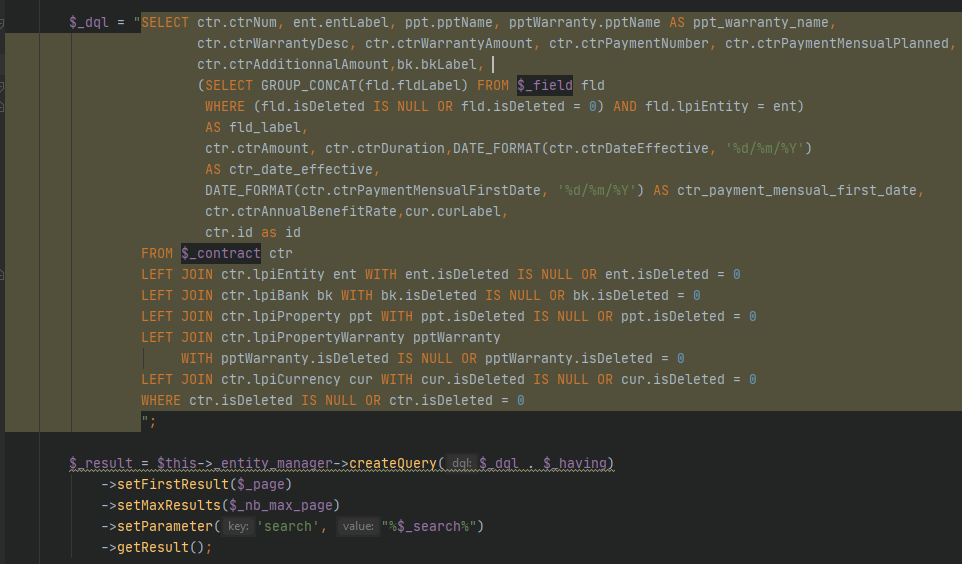
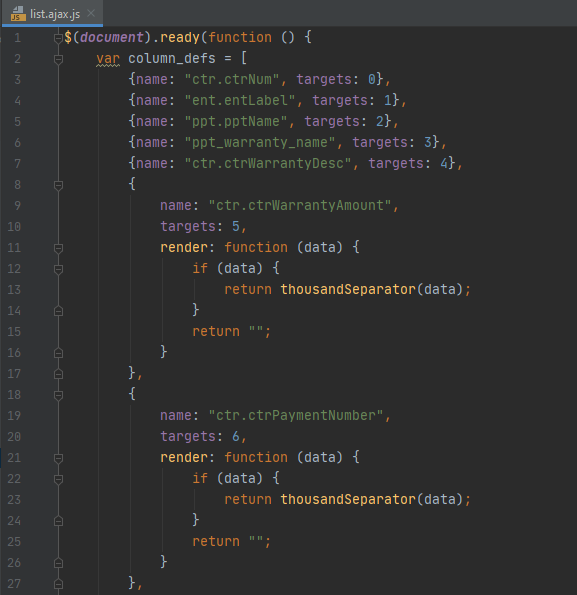




Figure 51 : Repository du contrat

La figure 52 montre le script du contrat.



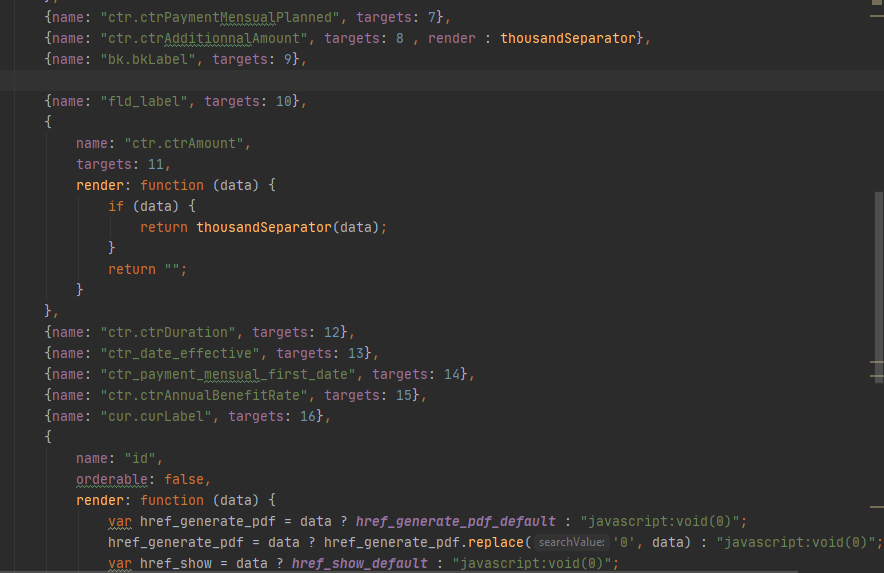


Figure 52 : Script du contrat

8.2.4 Présentation de l’application

La figure 53 montre la page d’authentification de la plateforme.

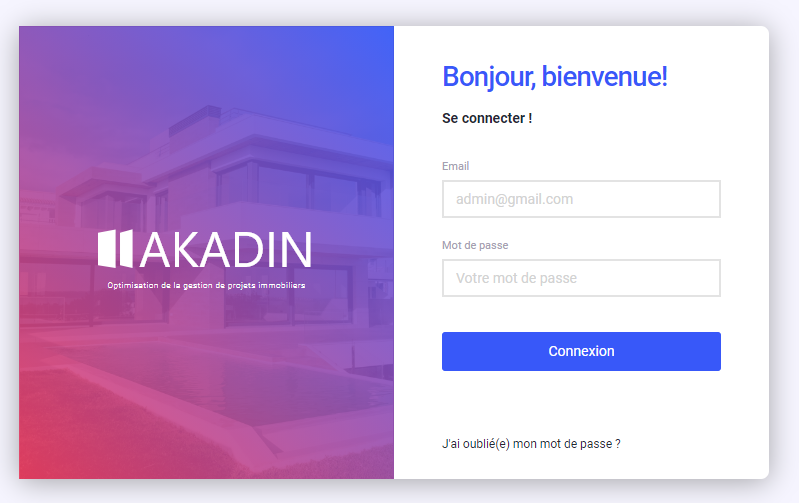


Figure 53 : Page d’authentification de la plateforme

La figure 54 montre la page d’accueil de la plateforme.

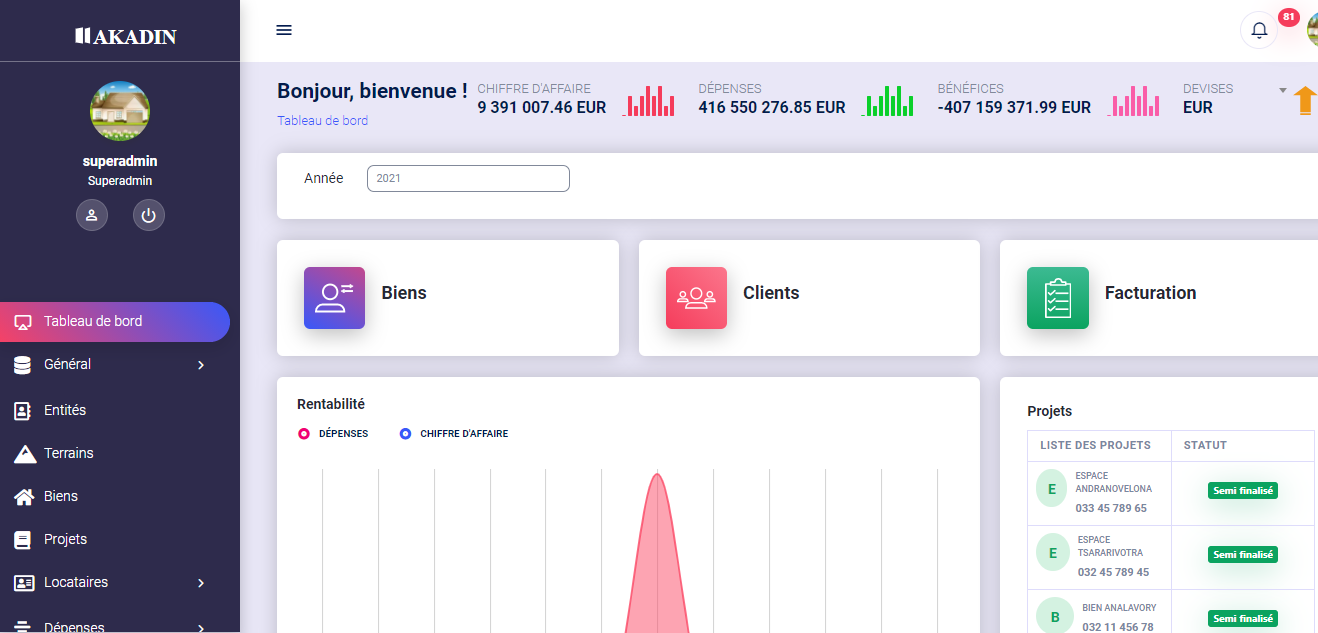


Figure 54 : Page d’accueil de la plateforme

La figure 53 représente la liste des banques.

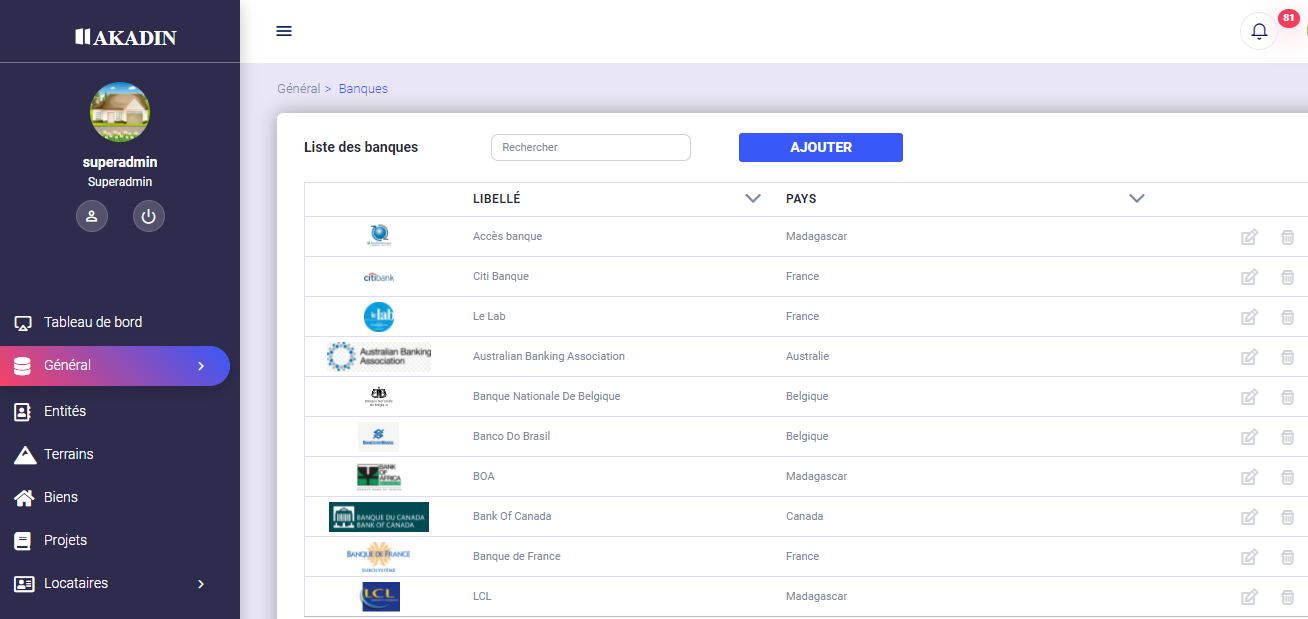


Figure 55 : Liste des banques

La figure représente la liste des devises.

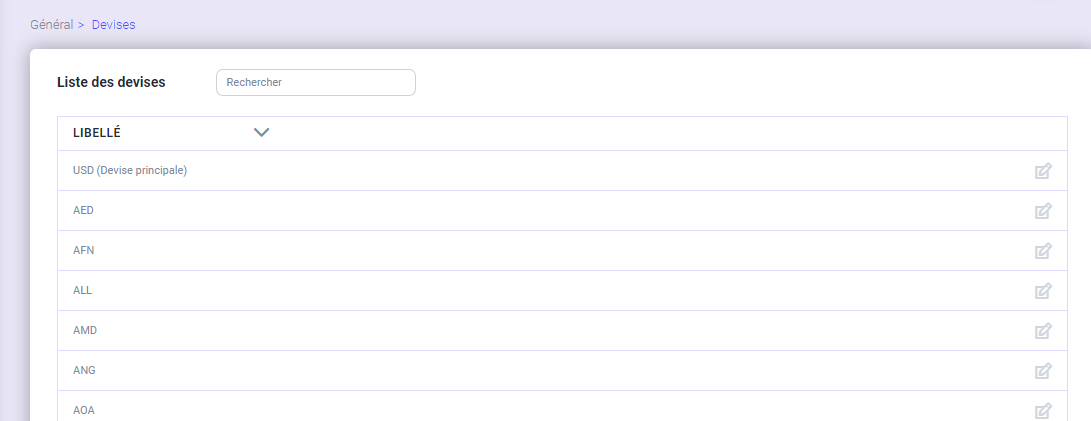


Figure 56 : Liste des devises

La figure 55 représente la liste des entités.

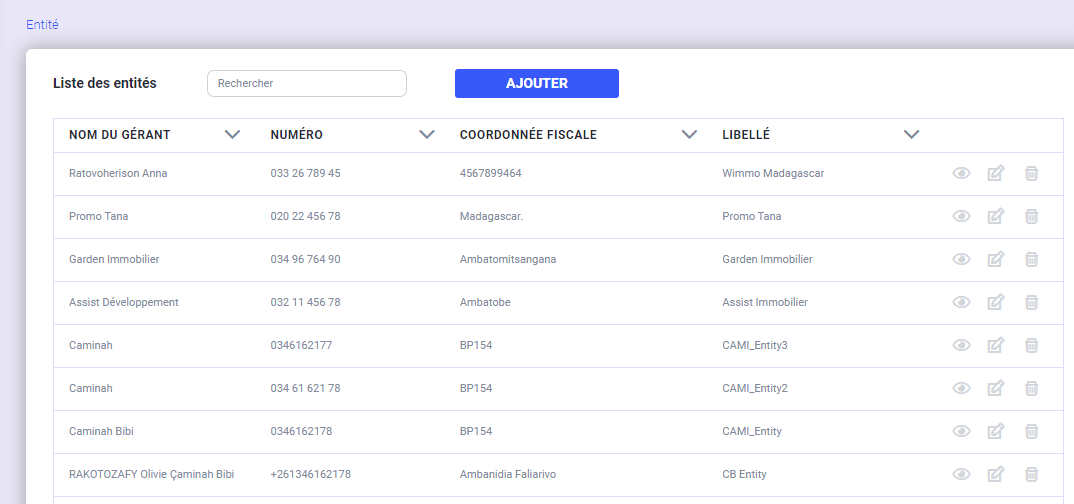


Figure 57 : Liste des entités

La figure 56 représente la liste des biens.



Figure 58 : Liste des biens

La figure 57 représente la liste des terrains.

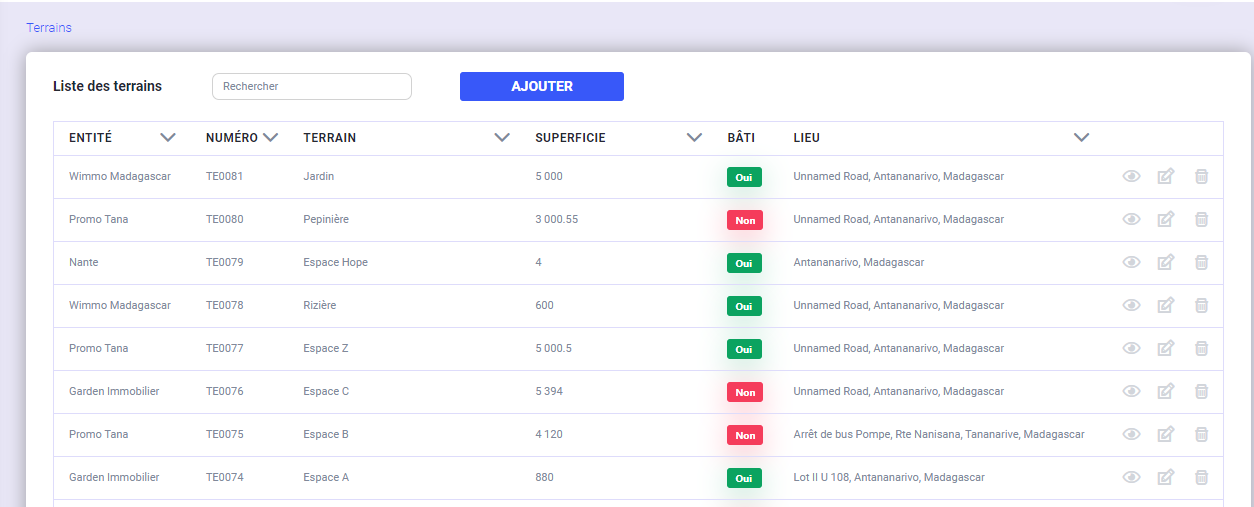


Figure 59 : Liste des terrains

La figure 58 représente la liste des agences.

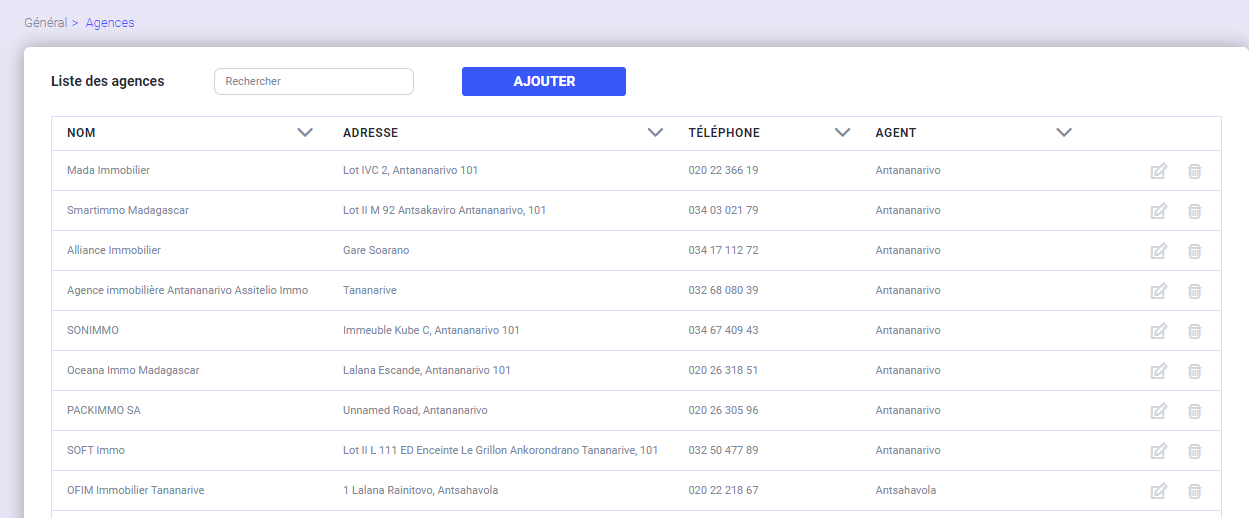


Figure 60 : Liste des agences

La figure 59 représente la liste des contrats.

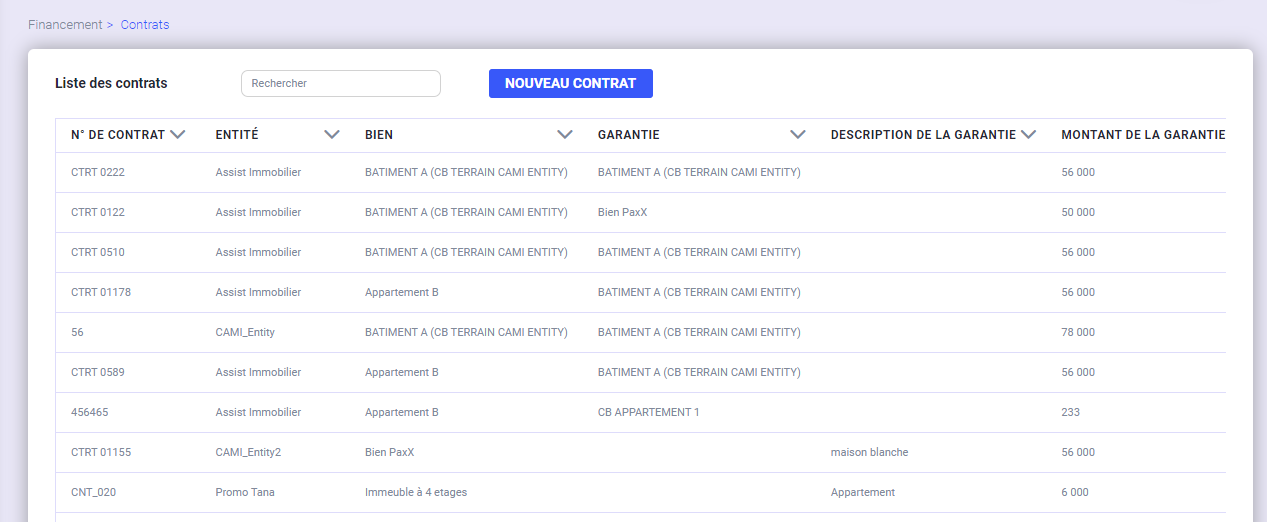


Figure 61 : Liste des contrats

CONCLUSION

Ce projet s’est déroulé au sein de la société LIVENEXX Antananarivo pour la conception et réalisation de la plateforme web pour la gestion d’un parc immobilier.

La conception et la réalisation du projet a nécessité une méthode de conception 2TUP, un Système de Gestion de Base de données MySQL, un langage de programmation PHP, un Framework PHP Symfony, d’un environnement de développement PHPStorm et d’un outil de versioning GIT.

La plateforme permet de gérer plusieurs modules tels que le financement, la ressource humaine, le stock, la comptabilité, la facturation et les rapports.

Cette plateforme est opérationnelle et répond aux besoins de l’entreprise AKADIN.

BIBLIOGRAPHIE

[1]. Joseph Gabay et David Gabay, « UML 2 ANALYSE ET CONCEPTION », Edition DUNOD en 2007

[2]. Pascal Roques, « UML 2 par la pratique, Etudes de cas et exercices corrigés », Edition Eyrolles, 6ème édition en 2007

[3]. Laurent Audibert, « UML 2, De l’apprentissage à la pratique », Edition Ellipses en 2013. [4]. Manuelle Lorenzo, apprendre la méthode 2TUP, Eyrolles, Paris 2013.

WEBOGRAPHIE

<https://symfony.com/doc/current/index.html>

<https://datatables.net/manual/>

<https://twig.symfony.com/doc/2.x/>

<https://getbootstrap.com/docs/4.1/getting-started/introduction/>

<https://www.w3schools.com/php/php_regex.asp>

<https://www.w3schools.com/jquery/jquery_ref_overview.asp>

<https://www.w3schools.com/sql/default.asp>

TABLE DE MATIERE

[CURRICULUM VITAE I](#_Toc80539594)

[SOMMAIRE GENERALE III](#_Toc80539595)

[REMERCIEMENTS IV](#_Toc80539596)

[LISTE DES FIGURES V](#_Toc80539597)

[LISTE DES TABLEAUX VII](#_Toc80539598)

[**LISTE DES ABREVIATIONS** VIII](#_Toc80539599)

[INTRODUCTION GENERALE 1](#_Toc80539600)

[Partie I : PRESENTATIONS GENERALE 2](#_Toc80539601)

[**Chapitre 1 : PRESENTATION DE L’ECOLE NATIONALE D’INFORMATIQUE** 3](#_Toc80539602)

[1.1 Informations d’ordre général 3](#_Toc80539603)

[1.2 Missions et historique 3](#_Toc80539604)

[1.3 Organigramme institutionnel de l’ENI 5](#_Toc80539605)

[1.4 Domaines de spécialisation 7](#_Toc80539606)

[1.5 Architecture des formations pédagogiques 7](#_Toc80539607)

[1.6 Relations de l’ENI avec les entreprises et les organismes 10](#_Toc80539608)

[1.7 Partenariat au niveau international 11](#_Toc80539609)

[1.8 Débouche professionnels des diplômes 13](#_Toc80539610)

[1.9 Ressources humaines 15](#_Toc80539611)

[Chapitre 2 : Présentation de LIVDHASBJF 17](#_Toc80539612)

[2.1Historique et missions 17](#_Toc80539613)

[2.2. Organisation générale 19](#_Toc80539614)

[**2.3. Processus de développement** 20](#_Toc80539615)

[Chapitre 3 : Description du projet 21](#_Toc80539616)

[3.1. Formulation 21](#_Toc80539617)

[**3.2. Objectifs et besoins de l’utilisateur** 21](#_Toc80539618)

[**3.3. Moyens nécessaires à la réalisation du projet** 22](#_Toc80539619)

[Partie II : ANALYSE ET CONCEPTION 23](#_Toc80539620)

[Chapitre 4. Analyse préalable 24](#_Toc80539621)

[4.1. Analyse de l’existant 24](#_Toc80539622)

[4.1.1. Organisation actuelle 24](#_Toc80539623)

[4.1.2. Inventaire des moyens matériels et logiciels 24](#_Toc80539624)

[4.2. Critique de l'existant 26](#_Toc80539625)

[4.3. Conception avant-projet 26](#_Toc80539626)

[4.3.1. Proposition des solutions 26](#_Toc80539627)

[4.3.2. Solution retenue 27](#_Toc80539628)

[**a.** **Choix de l’outil de conception** 32](#_Toc80539629)

[b. Choix du langage de Programmation 33](#_Toc80539630)

[c. Choix du Système de Gestion de Base de Données (SGBD) 34](#_Toc80539631)

[d. Choix du Framework PHP 36](#_Toc80539632)

[Chapitre 5 : Analyse conceptuelle 38](#_Toc80539633)

[5.1. Dictionnaire de données 38](#_Toc80539634)

[5.2. Règle de gestion 42](#_Toc80539635)

[Chapitre 6 : Analyse conceptuelle 43](#_Toc80539636)

[6.1. Diagramme de classe de conception 43](#_Toc80539637)

[6.1.1 Identification des acteurs 43](#_Toc80539638)

[6.2 Spécification des besoins fonctionnels 43](#_Toc80539639)

[6.2.1 Identification des cas d’utilisation 43](#_Toc80539640)

[Gérer les statuts de bien 47](#_Toc80539641)

[Superadmin 47](#_Toc80539642)

[Demande d’ajouter un statut de bien 47](#_Toc80539643)

[Résultat de la demande d’ajout d’un statut de bien 47](#_Toc80539644)

[Demande de modifier un statut de bien 47](#_Toc80539645)

[Résultat de la demande de modification d’un statut de bien 47](#_Toc80539646)

[Demande de supprimer un statut de bien 47](#_Toc80539647)

[Résultat de la demande de suppression d’un statut de bien 47](#_Toc80539648)

[Gérer les biens 47](#_Toc80539649)

[Superadmin 47](#_Toc80539650)

[Demande d’ajouter un bien 47](#_Toc80539651)

[Résultat de la demande d’ajout d’un statut de bien 47](#_Toc80539652)

[Demande de modifier un bien 47](#_Toc80539653)

[Résultat de la demande de modification d’un bien 47](#_Toc80539654)

[Demande de supprimer un bien 47](#_Toc80539655)

[Résultat de la demande de suppression d’un bien 47](#_Toc80539656)

[6.3 Diagramme de cas d’utilisation 49](#_Toc80539657)

[6.4 Diagramme de séquence système 51](#_Toc80539658)

[6.5 Spécification des besoins techniques 62](#_Toc80539659)

[6.5.1 Capture de besoins techniques 62](#_Toc80539660)

[6.5.2 Diagramme de cas d’utilisation technique : 62](#_Toc80539661)

[6.6 Modélisation du domaine 63](#_Toc80539662)

[Chapitre 7 : Conception détaillée 65](#_Toc80539663)

[7.1 Architecture du Système 65](#_Toc80539664)

[7.2 Diagramme de séquence de conception pour chaque cas d'utilisation 66](#_Toc80539665)

[7.3 Diagramme de classe de conception pour chaque cas d'utilisation. 77](#_Toc80539666)

[7.4 Diagramme de classe de conception 79](#_Toc80539667)

[7.5 Diagramme de paquetage 80](#_Toc80539668)

[7.6 Diagramme de déploiement 81](#_Toc80539669)

[Chapitre 8 : Mise en place de l’environnement de développement 83](#_Toc80539670)

[8.1. Installation et configuration des outils 83](#_Toc80539671)

[8.1.1. Installation de Xampp 83](#_Toc80539672)

[8.1.2. Installation de PHPStorm 84](#_Toc80539673)

[8.2. Développement de l’application 86](#_Toc80539674)

[8.2.1. Création de la base de données 86](#_Toc80539675)

[8.2.3. Codage de la plateforme 87](#_Toc80539676)

[8.2.4 Présentation de l’application 95](#_Toc80539677)

[CONCLUSION 100](#_Toc80539678)

[BIBLIOGRAPHIE X](#_Toc80539679)

[WEBOGRAPHIE X](#_Toc80539680)

[TABLE DE MATIERE XI](#_Toc80539681)

RESUME

Dans ce projet, nous avons conçu et réalisé une plateforme web de gestion de parc immobilier au sein du Société LIVENEXX pour l’entreprise AKADIN.Ce plateforme a pour but de facilite la gestion de contrat lie aux biens, aux entités et aux garanties. Pour réaliser ce projet, on a utilisé un outil de versioning GIT, un outil de modélisation Visual Paradigm,un Système de Gestion de Base de Donnée MySQL, un environnement de développement PhpStorm,la langage de programmation PHP, le Framework Symfony et la méthode de conception 2TUP qui est pilote par UML.

L’application répond aux attentes des utilisateurs grâce à ces fonctionnalités

Mots clés : Akadin - Symfony - Git - UML

ABSTRACT

In this project, we designed and created a web platform for real estate management within the LIVENEXX Company for the AKADIN Company. This platform aims to facilitate the management of contracts related to goods, entities and guarantees. To carry out this project, we used a GIT versioning tool, a Visual Paradigm modeling tool, a MySQL Database Management System, PhpStorm development environment, the PHP programming language, the Symfony framework and the design method 2TUP which is piloted by UML.

The application meets user expectations thanks to these features

Keywords: Akadin - Symfony - Git - UML