



REPUBLIQUE DU BENIN



MINISTRE DE L'ENSEIGNEMENT SUPERIEUR

UNIVERSITE D'ABOMEY-CALAVI

THE ABDUS SALAM INTERNATIONAL CENTER FOR THEORETICAL
PHYSICS

INSTITUT DE MATHEMATIQUES ET DE SCIENCES PHYSIQUES
(IMSP)

Mémoire de fin de formation pour l'obtention du diplôme de
Licence Professionnelle

Option : INFORMATIQUE

THEME :

CONCEPTION D'UNE
APPLICATION WEB DE GESTION
IMMOBILIERE

Présenté par :

YANCLO Mahougnon Fifa Paule Ursula

Sous la supervision du :

Dr.HOUNGUE Pélagie,

Enseignante - chercheur à l'IMSP-UAC

Année universitaire : 2021-2022

Je dédie ce travail à :

- ★ Dieu tout puissant, pour cette grâce qu'il m'accorde.
- ★ Mon grand-père Carlos Juliano OMOWEWE qui m'a toujours soutenu dans ses prières.
- ★ Mes chers parents Raymond YANCLO et Cécile NOUATIN qui ont été mes guides sur cette terre.
- ★ Mon frère YANCLO Jean-Albin, et mes sœurs YANCLO Raphaella et YANCLO Luz pour leurs encouragements et soutiens.
- ★ Mes amis.

REMERCIEMENTS

Je remercie le Seigneur qui m'a a gardé, protégé et assisté tout au long de cette année universitaire.

Mes sincères remerciements à mon Encadreur, Dr Pélagie HOUNGUE, pour sa disponibilité, ses remarques et conseils qui ont beaucoup contribué à l'amélioration de ce travail.

Mes remerciements s'adressent aussi au responsable de la formation des Classes Préparatoires aux Grandes Écoles de l'Institut de Mathématiques et de Sciences Physiques (IMSP), Professeur Carlos OGOUYANDJOU pour ces prodigieux conseils.

Mes sincères remerciements à tous mes Enseignants pour avoir partagé avec nous sans retenue leurs connaissances.

Je remercie particulièrement mes parents pour leur inébranlable soutien et pour la confiance qu'ils ont en moi.

Je ne saurai terminer cette liste sans remercier tous mes amis principalement Fawaz CHABI BOUKARI et David DOSSEH, mes camarades et toutes les personnes qui de près ou de loin m'ont aidé dans la réalisation de ce projet.

SOMMAIRE

DÉDICACES	i
REMERCIEMENTS	ii
ACRONYMES	iv
Tables des figures	v
RÉSUMÉ	vi
ABSTRACT	vii
INTRODUCTION GÉNÉRALE	1
1 ANALYSE ET CONCEPTION	2
1.1 Introduction	2
1.2 Diagramme de cas d'utilisation	2
1.3 Description textuelle de quelques Cas d'utilisation	5
1.4 Diagramme de séquence de quelque cas d'utilisations	8
1.5 Diagramme de classe :	13
1.6 Conclusion	16
2 MISE EN ŒUVRE ET RÉALISATION	17
2.1 Introduction	17
2.2 Langage de programmation	17
2.3 Framework, Serveur et Base de données	18
2.4 Quelques interfaces de notre application	18
2.5 Conclusion	27
CONCLUSION GÉNÉRALE	28
Bibliographie	29
Annexes	32

- [ORM] : Object Relational Mapping.
- [SGBD] : Système de Gestion de Base de Données.
- [SPA] : Single Page Application.
- [UML] : Unified Modeling Language.

Liste des figures

1.1	Diagramme de cas d'utilisation	4
1.2	Diagramme de séquence : Réserver bien	10
1.3	Diagramme de séquence : Effectuer un paiement	12
1.4	Diagramme de Séquence : Faire des commentaires	14
1.5	Diagramme de classe	15
2.1	page d'accueil de notre application(1)	19
2.2	page d'accueil de notre application(2)	20
2.3	page d'achat	21
2.4	page de connexion	22
2.5	page de d'inscription	23
2.6	page de location	24
2.7	page de réservation	25
2.8	informations d'un bien	26

La gestion immobilière est un domaine qui requiert beaucoup de polyvalence et un énorme sens de l'organisation. Le quotidien d'un gestionnaire immobilier consiste à gérer et administrer une foule de documents. En plus de veiller à la satisfaction et au bon déroulement des activités de tout autre individu qui interagit avec les immeubles à revenus. En ce sens, l'organisation devient chose primordiale et l'économie de temps et d'énergie devient un facteur déterminant. C'est pourquoi aujourd'hui, il existe plusieurs applications web pour résoudre ce problème. Allant dans ce même sens, nous avons voulu concevoir une application web de gestion immobilière notre part de solution aux différents problèmes qui y sont liés. L'exécution et l'élaboration du projet ont été basées sur l'approche Orienté Objet, Posgresql a été utilisé pour le Système de Gestion de Base de Donnée. Typeorm nous a aidé à générer la base de donnée à partir des classes créées dans les fichiers modèles de notre projet. La plateforme se doit de permettre à de potentiels propriétaires de biens immobiliers d'exposer ces derniers (descriptions et caractéristiques), de trouver des acheteurs et des locataires. Les acheteurs ou les locataires ont la possibilité de consulter des détails concernant les biens qui les intéressent en ligne et d'effectuer des réservations.

Mots clés : propriétaire, biens immobiliers, client, locataire, demande de bien, location, achat, vente, plateforme de gestion immobilière

Property management is a job that requires a lot of versatility and a huge sense of organization. The daily life of a property manager consists of managing and administering a host of documents. in addition to ensuring the satisfaction and smooth running of the activities of any other individual who interacts with income properties. In this sense, organization becomes essential and saving time and energy becomes a determining factor. That is why today there are several web applications to solve this problem. Going in the same direction, we wanted to design a web application for property management, our part of the solution to the various problems associated with it. The execution and development of the project were based on the Object Oriented approach, Posgresql was used for the System of Database Management. Typeorm helped us generate the database from the classes created in the model files of our project. The platform must allow potential owners properties to expose them (descriptions and characteristics), to find buyers and tenants. Buyers or tenants have the option to view details concerning the goods that interest them online and to carry out reservations.

Keywords : owner, real estate, client, tenant, property request, rental, purchase, sale, property management platform

INTRODUCTION GÉNÉRALE

De nos jours, les bailleurs sont confrontés à plusieurs problèmes dans la gestion de leur biens immobiliers tels que l'assurance loyers impayés, la recherche locataire, la rédaction des baux, le reversement des loyers, la gestion des charges, l'administratif d'un point de vue général ainsi que le juridique. Les clients de leurs côtés ont aussi la difficulté de rechercher des logements et tout autres biens immobiliers. Nous avons donc décidé de mettre en place une plateforme de gestion immobilière pour pouvoir aider les propriétaires qui peuvent désormais suivre l'évolution des choses directement grâce à un espace personnel dédié, sur notre site et les clients qui peuvent aussi rapidement trouver réponse à leur quête. La présentation est subdivisée en deux parties, la partie d'analyse et conception qui nous permet de présenter les technologies qui nous ont permis de mettre en place le système et la partie de réalisation et mise en œuvre qui nous présente les interfaces, la base de donnée et quelques scripts utiles pour la réalisation du projet.

ANALYSE ET CONCEPTION

1.1 Introduction

Dans ce chapitre, nous abordons l'étude détaillée de notre solution à la problématique posée. Notre étude est basée sur une modélisation UML (Unified Modeling Language ou Langage de Modélisation Unifié) car il intègre le concept d'objet et les principes associés à l'objet notamment l'héritage, la composition et la transition d'état et de sa manière visuelle et graphique de décrire les besoins. Notre choix s'est porté sur Modelio comme outils de modélisation car il permet une modélisation extrêmement complète et performante. Nous présentons les différents diagrammes UML qui nous ont permis de décrire notre étude. Cette partie s'articule autour de quatre(4) sous chapitres.

1.2 Diagramme de cas d'utilisation

1.2.1 Définition

Les diagrammes de cas d'utilisation permettent de décrire les grandes fonctionnalités du Système du point de vue des utilisateurs. Un cas d'utilisation est un service rendu par le système à un acteur.

1.2.2 Rôle du diagramme de cas d'utilisations

- Donne une vue du système dans son environnement extérieur.
- Définit la relation entre l'utilisateur et les éléments que le système met en oeuvre.

1.2.3 Identification des acteurs

Les principaux profils qui auront à utiliser l'application web sont les suivants :

— **Administrateur**

- Gère les droits sur les différents acteurs.
- Il gère les types de biens qui seront sur l'application.
- Il gère également les villes dans lesquels se trouvent les biens.

— **Utilisateurs**

- Il consulte les biens.
- Il effectue une réservation pour louer un bien.
- Il achète les biens
- Il fait une location des biens

— **Propriétaire**

- Il crée les biens
- Il supprime les biens
- Il met les biens à jours
- Il recherche les biens

1.2.4 Diagramme de cas d'utilisation

La figure [1.1](#) représente le diagramme des Cas d'Utilisation :

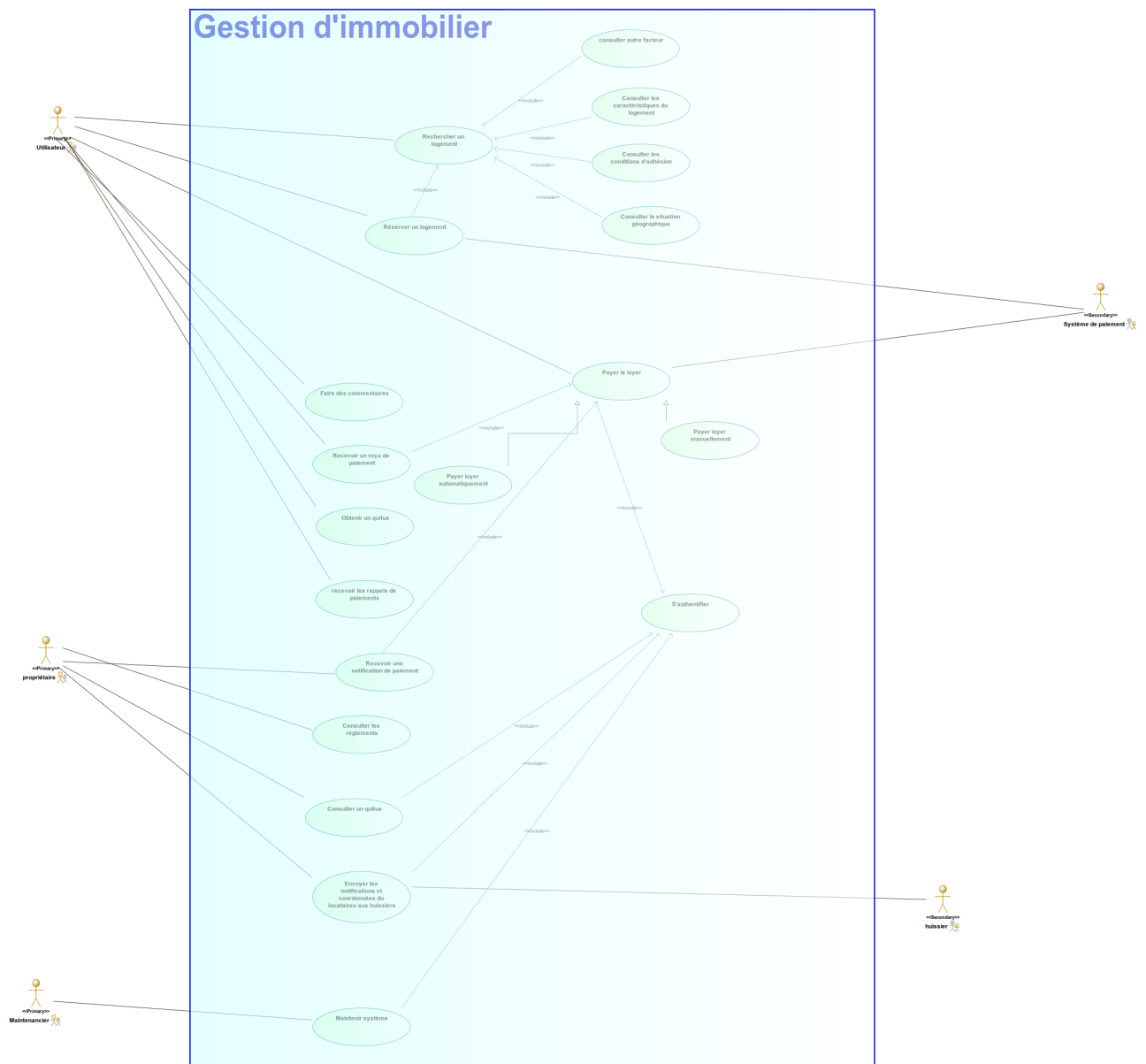


FIGURE 1.1 – Diagramme de cas d'utilisation

1.3 Description textuelle de quelques Cas d'utilisation

1.3.1 Cas d'utilisation : Effectuer un paiement et Recevoir un reçu de paiement

▲ Identification

Nom : recevoir un reçu de paiement

But : permettre aux clients et aux propriétaires de recevoir un reçu de paiement

Acteurs principaux : clients, propriétaires

Acteurs secondaires : Système de paiement

Date : 26/07/22

Responsable : Yanclo Mahougnon Fifa Paule Ursula

Version : 3.0

▲ Pré-conditions

Le site est accessible et opérationnel

L'internaute dispose d'accès à internet

▲ Scénario nominal

1. Le système demande au système de paiement d'afficher un formulaire de paiement
2. Le système de paiement affiche un formulaire de paiement
3. Le client remplit et soumet le formulaire de paiement
4. Le système de paiement traite les informations du formulaire de paiement
5. Le système de paiement envoie une réponse au système
6. Le système de paiement envoie le reçu de paiement au client et au propriétaire du bien

▲ Enchaînement alternatifs :

▲ Enchaînements d'erreurs :

E1 :Le système de paiement envoie une réponse négative

L'enchaînement démarre au point 5 du scénario nominal
6 :Le système de paiement notifie aux client que le paiement n'a pas abouti
Le cas d'utilisation se termine avec échec.

▲ Post-conditions :

le paiement a été effectué avec succès et le système de paiement a envoyé un reçu au client et au propriétaire du bien.

1.3.2 Cas d'utilisation : Faire des commentaires

▲ Identification

Nom : :faire des commentaires

But : permettre aux clients de donner leur avis sur un bien auquel ils ont déjà accédé une fois.

Acteurs principaux : clients

Acteurs secondaires : Système

Date : 26/07/22

Responsable : Yanclo Mahougnon Fifa Paule Ursula

Version : 3.0

▲ Pré-conditions

Le site est accessible et opérationnel

L'internaute dispose d'accès à internet

L'internaute a déjà accédé au bien en question au moins une fois.

▲ Scénario nominal

1. Le système affiche un formulaire d'identification
2. Le client remplit et soumet le formulaire
3. Le système traite les informations du formulaire
4. Le système envoie une réponse à l'internaute
5. Le système donne accès à l'utilisateur

▲ Enchaînement alternatifs :

A1 :Le formulaire d'identification est mal rempli

L'enchaînement démarre au point 3 du scénario nominal

4 :Le système notifie le(s) champ(s) mal rempli(s)

5 :L'utilisateur rectifie les erreurs notifiées par le système

L'enchaînement reprend au point 2

▲ Enchaînements d'erreurs :

E1 :L'internaute n'a jamais accédé au bien en question

L'enchaînement démarre au point 4 du scénario nominal

5 :Le système informe l'utilisateur de la raison pour laquelle il ne peut pas commenter le

Le cas d'utilisation se termine avec échec.

▲ Post-conditions :

le système donne accès à l'internaute.

1.3.3 Cas d'utilisation : Obtenir un quitus

▲ Identification

Nom : obtenir un quitus

But :permettre aux propriétaire de biens de donner leur avis sur leurs locataires

Acteurs principaux : locataires,propriétaires

Acteurs secondaires : Système

Date : 26/07/22

Responsable : Yanclo Mahougnon Fifa Paule Ursula

Version : 3.0

▲ Pré-conditions

Le site est accessible et opérationnel

L'internaute dispose d'accès à internet

▲ Scénario nominal

1. Le client fait une première demande(type de demande = location) de bien
2. Le propriétaire du bien accepte la demande
3. Le système traite les informations du formulaire
4. Le client paye la caution
5. Le système attribut un quitus au locataire
6. Le système attribut un quitus au locataire

▲ Enchaînement alternatifs :

A1 :la demande n'a pas abouti

L'enchaînement démarre au point 1 du scénario nominal

▲ Enchaînements d'erreurs :

E1 :Le propriétaire du bien refuse la demande

L'enchaînement démarre au point 2 du scénario nominal

5 :Le système notifie au client que sa demande a été refusée

Le cas d'utilisation se termine avec échec.

▲ Post-conditions :

Le client obtient son quitus.

1.4 Diagramme de séquence de quelque cas d'utilisations

1.4.1 Définition

Les diagrammes de Séquences sont la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique dans le formulation Unified Modeling Language.

Diagramme de séquence : Réserver bien

Ce diagramme explique l'interaction entre le client et le système dans le cas d'une réservation de biens(appartement,une parcelle, bâtiment).

La figure 1.2 nous illustre ce diagramme

1.4.2 Diagramme de séquence : Effectuer un paiement

Ce diagramme explique l'interaction entre le client et le système de paiement lorsque ce dernier s'apprête à faire un achat ou payer un loyer (appartement, une parcelle, bâtiment).

La figure [1.3](#) nous illustre ce diagramme.

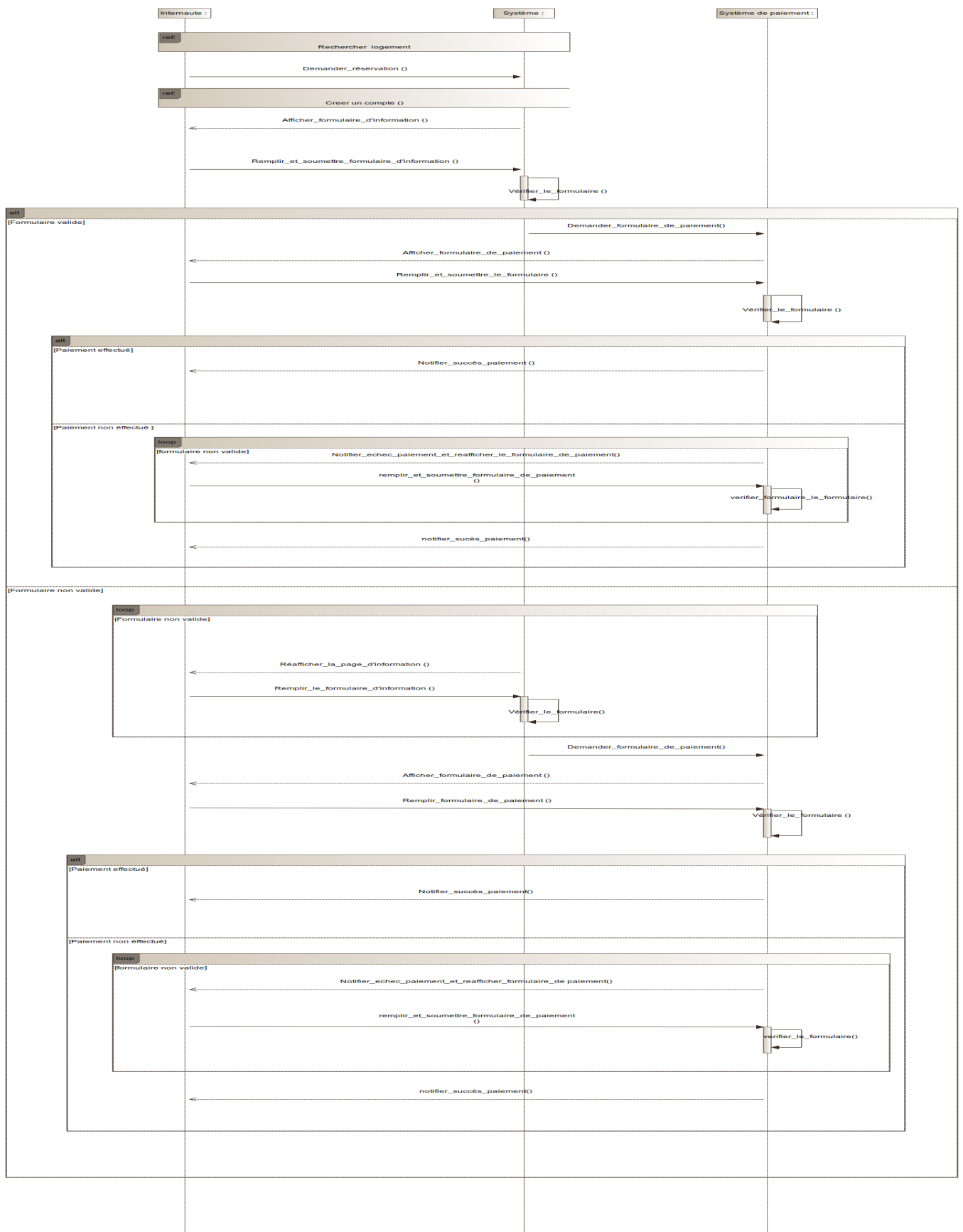


FIGURE 1.2 – Diagramme de séquence : Réserver bien

1.4.3 Diagramme de séquence : Faire des commentaires

Ce diagramme explique l'interaction entre le client et le système lorsque ce dernier s'apprête à commenter un bien sur la plate-forme (appartement, une parcelle, bâtiment).

La figure [1.4](#) nous illustre ce diagramme

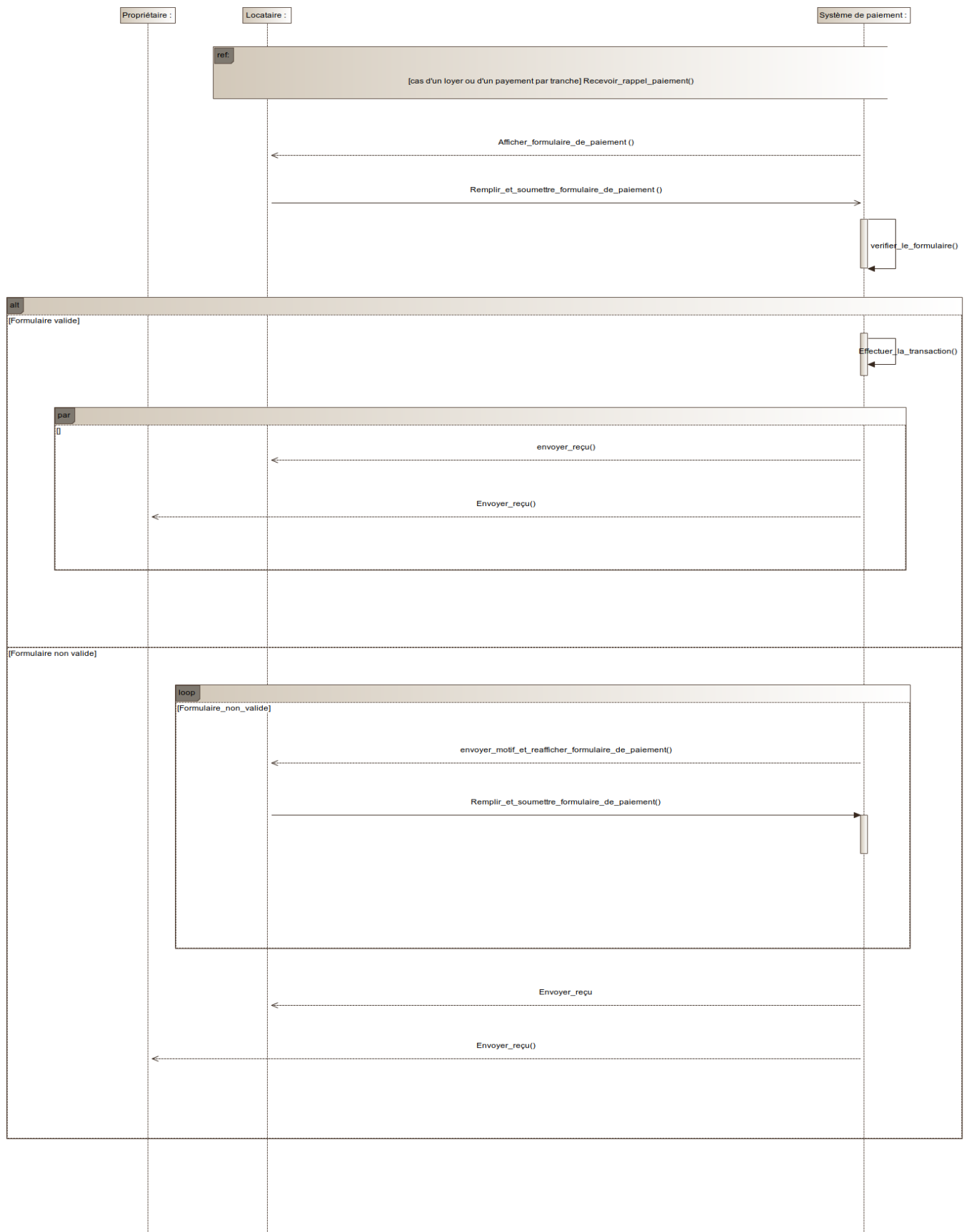


FIGURE 1.3 – Diagramme de séquence : Effectuer un paiement

1.5 Diagramme de classe :

Définition Le diagramme de classe représente bien la structure statique du code, par le biais des attributs et des relations entre les classes. C'est un diagramme principal qui est la vue de plus haut niveau avec l'ensemble des classes de l'application, il montre les briques de base statiques : classes, associations, interfaces, attributs, opérations...

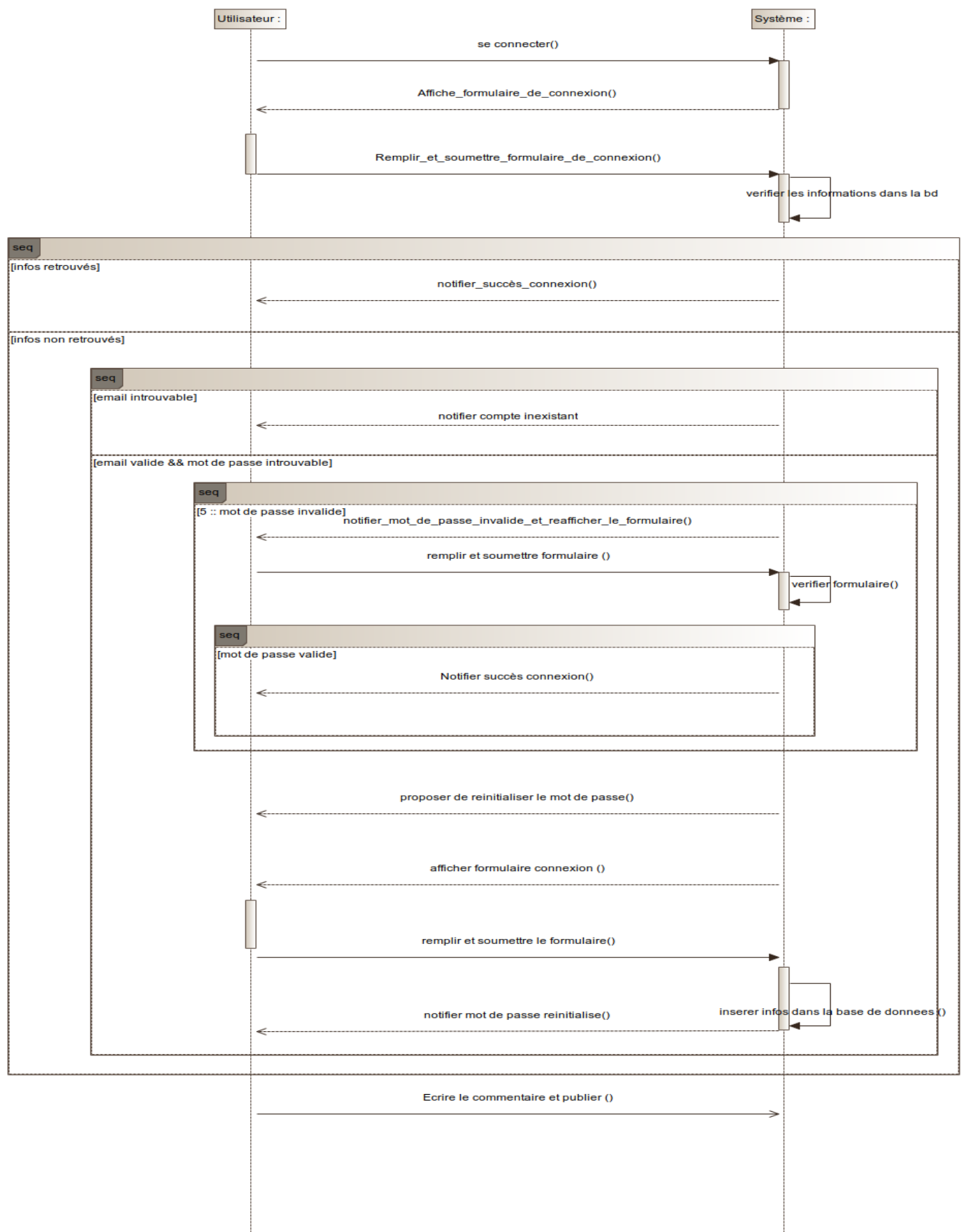


FIGURE 1.4 – Diagramme de Séquence : Faire des commentaires

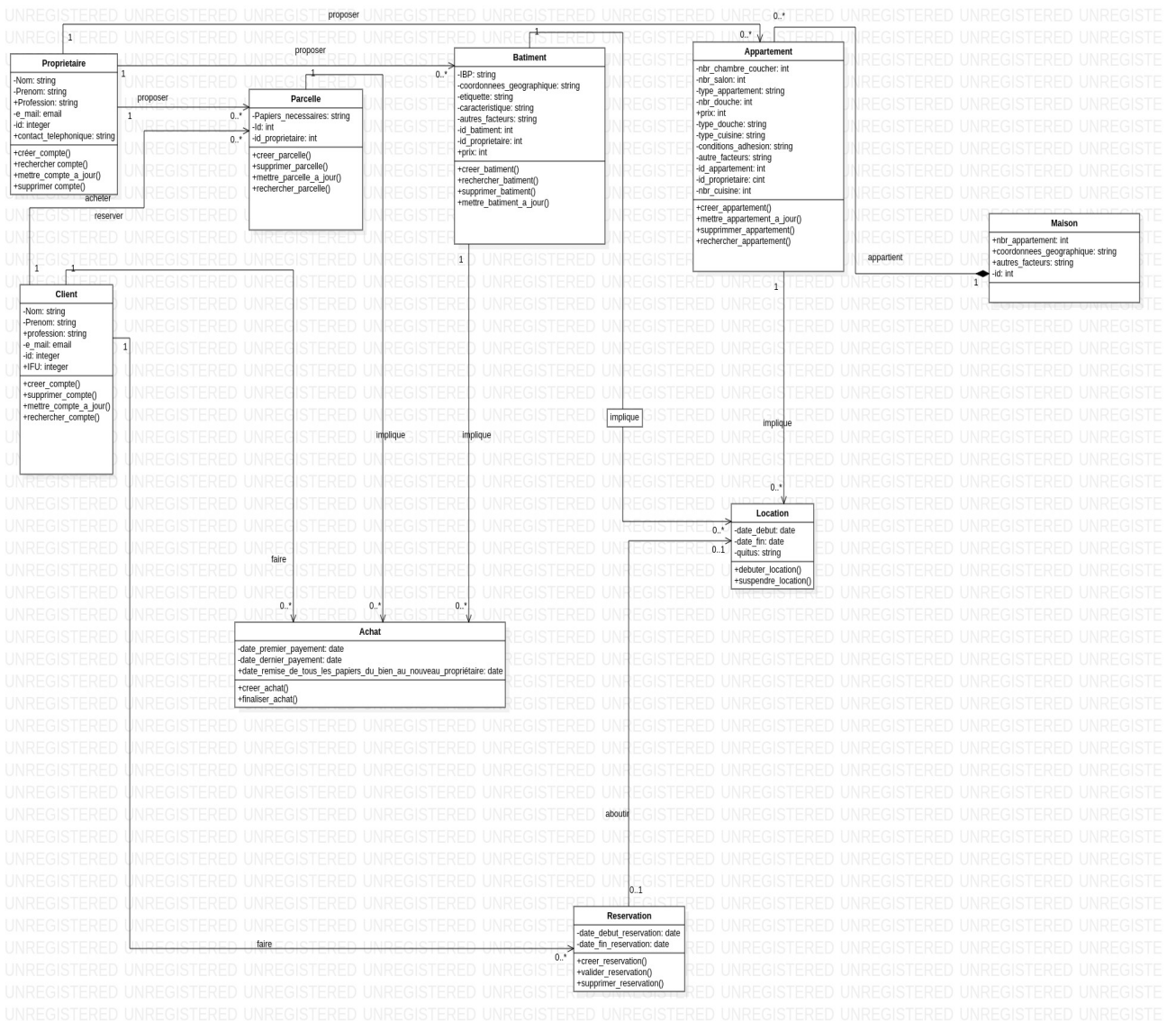


FIGURE 1.5 – Diagramme de classe

1.6 Conclusion

En somme, dans cette partie, après une analyse profonde, nous avons présenté le diagramme de cas d'utilisation, de classes et les diagrammes de séquences de notre plateforme. Dans le prochain chapitre, nous verrons les outils qui nous ont permis de réaliser cette application sans oublier de voir quelques interfaces de notre plateforme.

MISE EN ŒUVRE ET RÉALISATION

2.1 Introduction

Dans ce chapitre nous abordons la mise en œuvre de notre solution. Nous parlons principalement des logiciels et des langages de programmation utilisés pour la réalisation de notre application. Nous présentons également quelques interfaces de notre application.

2.2 Langage de programmation

- **Le Langage HTML5** : HyperText Markup Language, est la dernière révision majeure du HTML (format de données conçu pour représenter les pages web).
- **Le Langage CSS3** : Cascading Style Sheets, Les feuilles de style en cascade forment un langage informatique qui décrit la présentation des documents HTML et XML. Les standards définissant CSS sont publiés par le World Wide Web Consortium (W3C). Introduit au milieu des années 1990, CSS devient couramment utilisé dans la conception de sites web et bien pris en charge par les navigateurs web dans les années 2000.
- **Le langage JavaScript** est un langage de programmation de scripts principalement employé dans les pages web interactives.
- **Le langage Typescript** est un sur-ensemble syntaxique de JavaScript qui ajoute un typage statique. Cela signifie essentiellement que TypeScript ajoute de la syntaxe au-dessus de JavaScript, ce qui permet aux développeurs d'ajouter des types.

2.3 Framework, Serveur et Base de données

- Le Serveur a été implémenter avec Node.js qui est un environnement d'exécution qui est utilisé pour exécuter JavaScript en dehors du navigateur. C'est le runtime qui permet d'écrire toutes nos tâches côté serveur, en JavaScript, telles que la logique métier, la persistance des données et la sécurité. Il ajoute également des fonctionnalités que le JavaScript du navigateur standard ne possède pas, comme par exemple l'accès au système de fichiers local.
- Mithril.js est un framework JavaScript moderne côté client pour la création d'applications à page unique(SPA). Il est petit (< 10 Ko gzip), rapide et fournit des utilitaires de routage et XHR prêts à l'emploi.
- Le Système de Gestion de Base de Données(SGBD) que nous avons utilisé est postgresSQL :
PostgreSQL est un système de gestion de base de données relationnelle orienté objet puissant et open source qui est capable de prendre en charge en toute sécurité les charges de travail de données les plus complexes. Alors que MySQL donne la priorité à l'évolutivité et aux performances, Postgres donne la priorité à la conformité et à l'extensibilité SQL.

2.4 Quelques interfaces de notre application

Voici l'interface de la page d'accueil de notre application. sur cette page se défilent les images correspondant au différents biens qui sont disponible.

En cliquant sur ces images, nous pouvons avoir des détails sur ces biens et les réserver si possible.

Les figure [2.1](#) et [2.2](#) représente la page d'accueil de notre application.

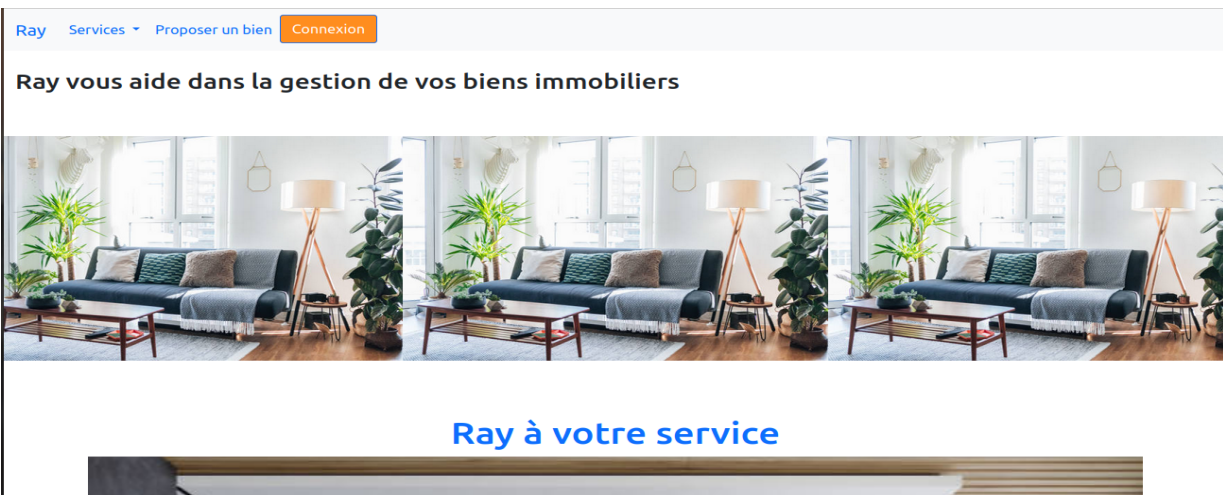


FIGURE 2.1 – page d'accueil de notre application(1)

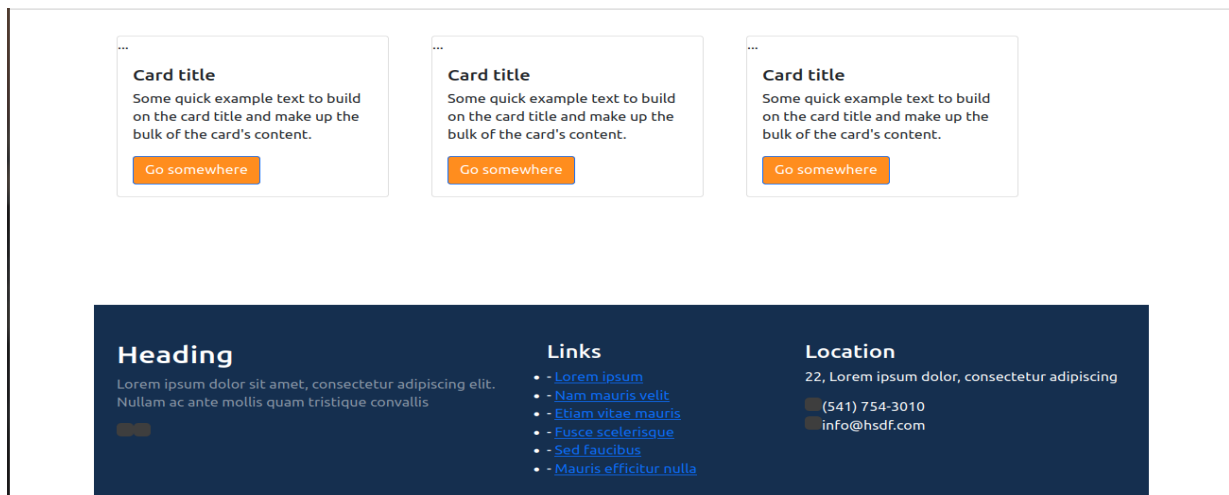


FIGURE 2.2 – page d'accueil de notre application(2)

La figure 2.3 représente la page d'achat.

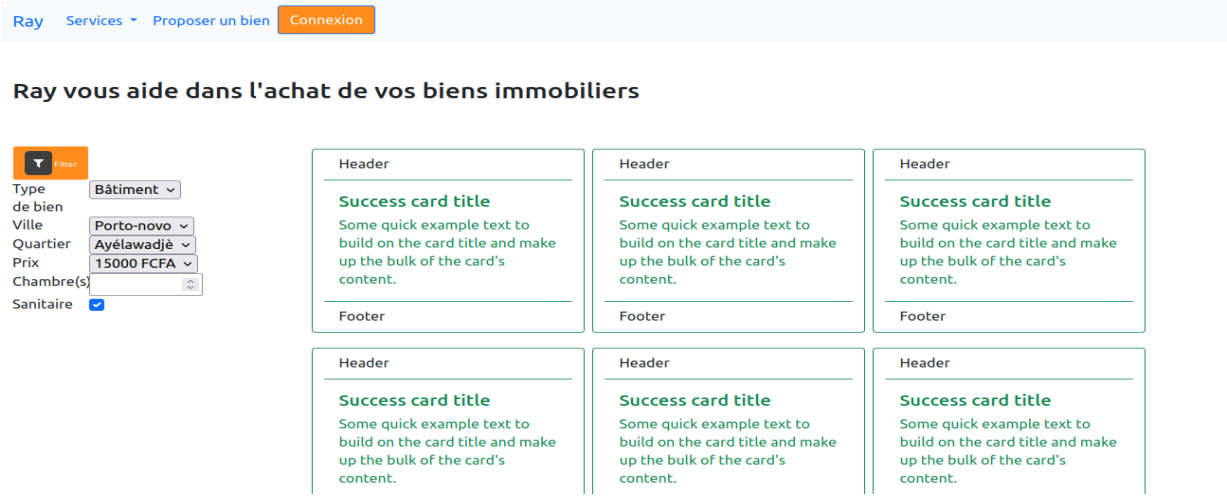


FIGURE 2.3 – page d'achat

La figure 2.4 représente la page de connexion. Sur cette page, le client saisit son email et son mot de passe.

Ray Services ▾ Proposer un bien Connexion

Email address

Password

☒ Remember me [Forgot password?](#)

Sign in

Not a member? [Register](#)

or sign up with:

FIGURE 2.4 – page de connexion

La figure 2.5 représente la page d’inscription. Sur cette page, le client remplit le formulaire dans l’objectif de créer un compte sur la plate-forme.

Ray

Services

Proposer un bien

Connexion

Email

Email

Password

Password

Address

1234 Main St

Profession

Votre métier

City

State

CLient

Zip

☐ Check me out

Sign in

FIGURE 2.5 – page de d’inscription

La figure 2.6 représente la page de location d'un bien.

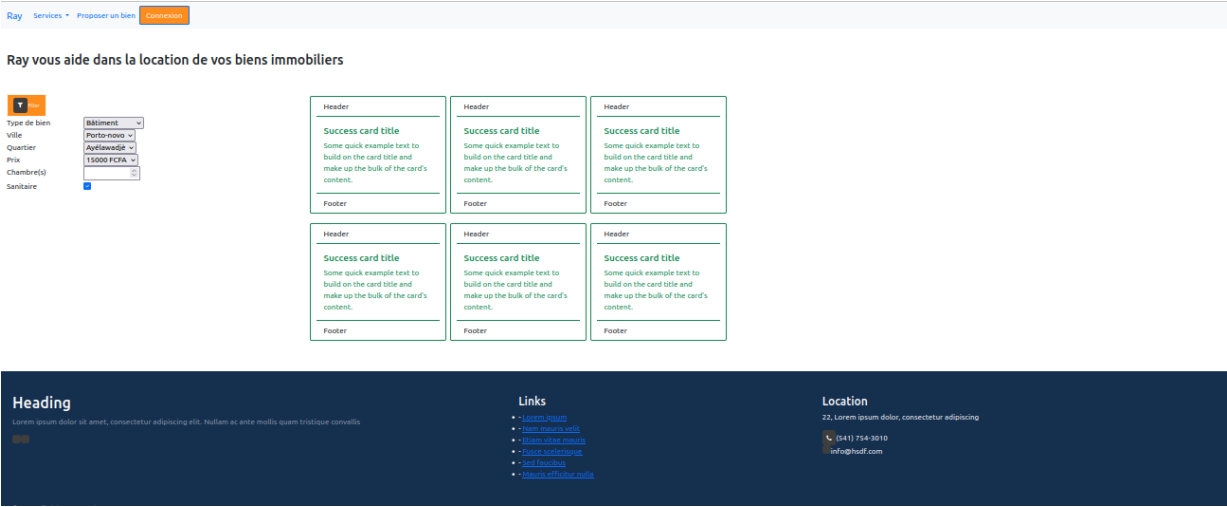


FIGURE 2.6 – page de location

La figure 2.7 représente la page de réservation d'un bien.

Ray

Services

Proposer un bien

Connexion

Ray vous aide dans la réservation de vos biens immobiliers

T

Ray

Type de bien

Ville

Quartier

Prix

Chambre(s)

Sanitaire

Appartement

Porto-novo

Ayélawadjé

15000 FCFA

Header

Success card title

Some quick example text to build on the card title and make up the bulk of the card's content.

Footer

Header

Success card title

Some quick example text to build on the card title and make up the bulk of the card's content.

Footer

Header

Success card title

Some quick example text to build on the card title and make up the bulk of the card's content.

Footer

Header

Success card title

Some quick example text to build on the card title and make up the bulk of the card's content.

Footer

Header

Success card title

Some quick example text to build on the card title and make up the bulk of the card's content.

Footer

Header

Success card title

Some quick example text to build on the card title and make up the bulk of the card's content.

Footer

Heading

22, Lorem ipsum dolor, consectetur adipiscing elit. Nullam ac ante mollis quam tristique convallis.

Links

• [Lorem ipsum](#)

• [Nam mauris velit](#)

• [Etiam vitae mauris](#)

• [Fusce scelerisque](#)

• [Sed faucibus](#)

Location

22, Lorem ipsum dolor, consectetur adipiscing

(541) 754-3010

info@hsdf.com

FIGURE 2.7 – page de réservation

La figure 2.8 représente les informations d'un bien.

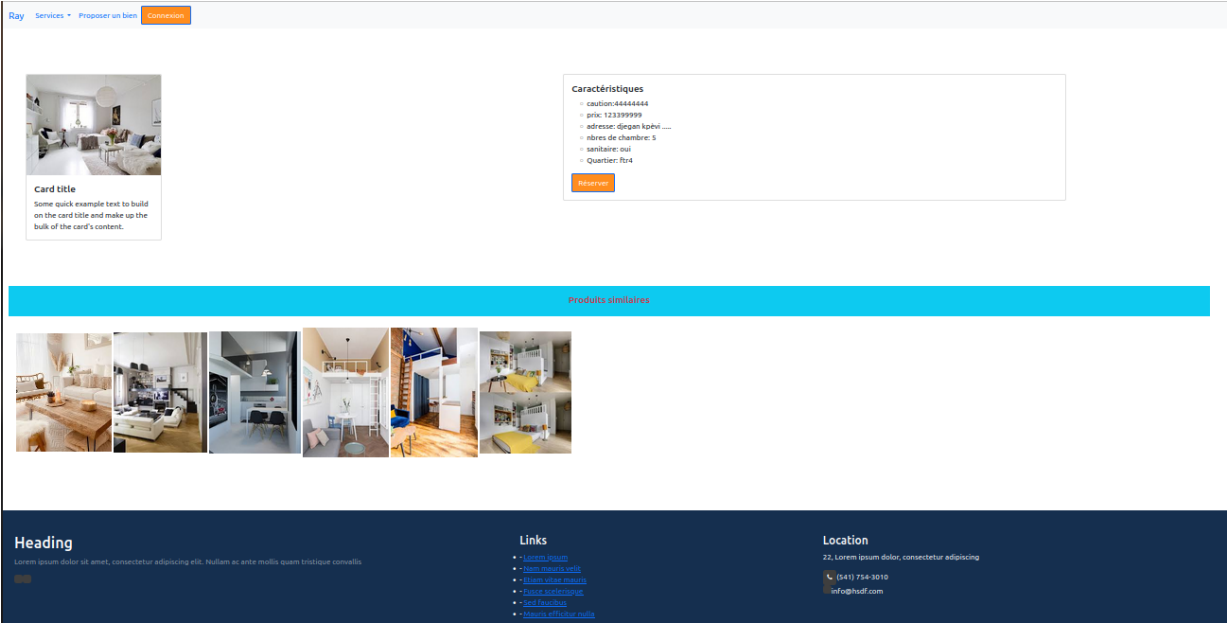


FIGURE 2.8 – informations d'un bien

2.5 Conclusion

En somme, dans cette partie, nous avons présenté les logiciels et les langages de programmation utilisés dans la réalisation de notre application. Nous avons également présenté quelques interfaces de notre plateforme.

CONCLUSION GÉNÉRALE

En définitive, notre projet nous a permis de réaliser une application web pour la gestion immobilière. Nous avons présenté le diagramme de cas d'utilisation afin d'exposer les différentes fonctionnalités de l'application web, nous avons élaboré le diagramme de classes qui représente la structure interne de l'application, ensuite vinrent les diagrammes de séquences qui soulignent les différentes interactions entre les utilisateurs et le système. Ensuite, nous avons présenté les logiciels et les langages de programmation utilisés sans oublier de présenter quelques interfaces de notre application. A travers ce projet, nous avons acquis beaucoup de connaissance en modélisation, et surtout dans la conception d'une application web. Cependant, nous n'avons pas pu aller au bout de la partie dans laquelle le client doit finaliser le paiement car nous n'avons pas pu nous rapprocher des institutions bancaires pour savoir un peu plus sur leurs systèmes de fonctionnement et de sécurité. Nous préconisons de faire une étude auprès de ces institutions bancaires afin de rendre plus efficace notre plateforme.

- [1] Cours de modélisation avec UML de Dr. Pélagie HOUNGUE.
- [2] Openclassrooms, Débutez l'analyse logicielle avec UML, 2018.
- [3] The Unified Modeling Language Reference Manual, J. Rumbaugh, I. Jacobson, G. Booch, Addison Wesley, Fall 98.
- [4] Modéliser objet avec UML, P.-A. Muller, N. Gaertner, Eyrolles, 2001.
- [5] Base de données : Les fondamentaux, Dr Hénoc SOUDE
- [6] C. S. Frédéric Brouard, Modélisation de bases de données : UML et les modèles entité-association, Eyrolles, 2015

Table des matières

DÉDICACES	i
REMERCIEMENTS	ii
ACRONYMES	iv
Tables des figures	v
RÉSUMÉ	vi
ABSTRACT	vii
INTRODUCTION GÉNÉRALE	1
1 ANALYSE ET CONCEPTION	2
1.1 Introduction	2
1.2 Diagramme de cas d'utilisation	2
1.2.1 Définition	2
1.2.2 Rôle du diagramme de cas d'utilisations	2
1.2.3 Identification des acteurs	2
1.2.4 Diagramme de cas d'utilisation	3
1.3 Description textuelle de quelques Cas d'utilisation	5
1.3.1 Cas d'utilisation : Effectuer un paiement et Recevoir un reçu de paiement	5
1.3.2 Cas d'utilisation : Faire des commentaires	6
1.3.3 Cas d'utilisation : Obtenir un quitus	7
1.4 Diagramme de séquence de quelque cas d'utilisations	8
1.4.1 Définition	8
1.4.2 Diagramme de séquence : Effectuer un paiement	9
1.4.3 Diagramme de séquence : Faire des commentaires	11
1.5 Diagramme de classe :	13
1.6 Conclusion	16
2 MISE EN ŒUVRE ET RÉALISATION	17
2.1 Introduction	17
2.2 Langage de programmation	17
2.3 Framework, Serveur et Base de données	18
2.4 Quelques interfaces de notre application	18
2.5 Conclusion	27
CONCLUSION GÉNÉRALE	28

Bibliographie	29
Annexes	32

- Le fichier `package.json` : Ce fichier contient diverses métadonnées relatives au projet. **Ce fichier est utilisé pour donner des informations à npm qui lui permettent d'identifier le projet ainsi que de gérer les dépendances du projet.**
- Les fichiers de quelques entités de notre base de données.

Généralement pour les fichiers `models`, nous avons utilisé `typeorm` pour pouvoir générer la base de donnée `postgresql` à partir des classes créées dans les fichiers `.ts`