

# **RAPPORT DU TEST**

- 
- **Conception et réalisation d'un site web (Immo`House)**
- 

**Encadré par :**

**LAAROUJA Chorouq**

**Encadré par :**

**MABED El khalil**

**M Vincent**

# TABLES DE MATIERES

## ANALYSE ET SPESIFICATION DES BESOINS

1 – Introduction

2 – Analyse globale de l’application

/Description des acteurs

/Spécification des besoins fonctionnels

/Spécification des besoins non fonctionnels

## CONCEPTION

1-Introduction

2-Conception générale

/Le langage UML

/ Avantages d'UML

/ Diagrammes d'UML utilisés

3- Conception détaillée

/ Diagramme de cas d'utilisation général

/ Description détaillé des cas d'utilisations

## REALISATION

1-Introduction

2 – Outils de développement

2.1- HTML

2.2- CSS

2.3-JavaScript

3– Interfaces de l'application

3.1- Page d'accueil

# ANALYSE ET SPECIFICATIONS DES BESOINS

## **1. Introduction :**

L'analyse et la spécification des besoins représentent la première phase du cycle de développement d'une application . Elle sert à identifier et décrire les acteurs réactifs du système, et à spécifier ensuite les différents besoins auxquels doit répondre l'application dans le but d'atteindre un résultat optimal et satisfaisant au client .

## **2. Analyse global de l'application :**

### **/ Description des acteurs :**

Un acteur représente un élément externe qui interagit avec le système. Cet élément peut être un utilisateur ou un système tiers. Tous les éléments extérieurs qui stimulent le système et tous les éléments extérieurs qui sont utilisés par le système sont représentés par des acteurs.

Les acteurs qui entrent en interaction avec notre système sont :

- **Utilisateur** : il s'agit d'un visiteur de notre site web
- **L'administrateur** : C'est la personne responsable de la gestion de notre système.

### **/ Spécification des besoins fonctionnels :**

Le système à réaliser doit satisfaire les exigences de la totalité des utilisateurs.

Nous présentons dans ce qui suit tous les besoins fonctionnels classés par acteur :

- L'utilisateur:

- Connexion
- Inscription
- Visiter les pages de notre site web

- L'administrateur :

- Gérer les comptes des utilisateurs (ajouter, supprimer ou modifier un utilisateur)

### **/ Spécification des besoins non fonctionnels :**

À part les besoins fondamentaux, notre système doit répondre aux critères suivants afin de respecter les exigences de l'utilisateur et de résister face aux risques de panne ou du non fonctionnement :

- **La performance** : Un site web doit être avant tout performant c'est-à-dire à travers ses fonctionnalités, répond à toutes les exigences des

usagers d'une manière optimale.

- **La convivialité** : l'application doit être facile à utiliser et à manipuler . En effet, les interfaces utilisateurs doivent être conviviales c'est-à-dire simples, ergonomiques et adaptées à n'importe quel utilisateur.
- **La maintenabilité** : le code de l'application doit être lisible et compréhensible afin d'assurer son état évolutif et extensible par rapport aux besoins .
- **La sécurité** : Notre application doit exiger l'établissement de la connexion avant l'autorisation de l'accès, n'accepter que les mots de passe répondant aux conditions de sécurité (longueur, caractères spéciaux...) et/ou se déconnecter automatiquement après temps morts d'inactivité afin de garantir la sécurité du système .

# CONCEPTION

## **1. Introduction :**

La conception a pour objectif de permettre la formalisation des étapes préliminaires du développement d'un système, afin de rendre ce développement plus fidèle aux besoins du client .

Cette phase permet de décrire de manière non ambiguë , le fonctionnement futur du système, afin d'en faciliter la réalisation .

Pour atteindre cet objectif, nous nous appuyons sur le formalisme UML basé sur les diagrammes, et offrant une flexibilité marquante.

Ainsi, dans la suite de ce chapitre, nous ciblons plus particulièrement la conception de notre base de données qui est considérée comme étant la tâche la plus ardue du processus de développement du système d'information .

## **2. Conception générale :**

### **/ Le langage UML :**

Notre choix est porté sur le langage UML puisqu'il convient pour toutes les méthodes objet et se prête bien à la représentation de l'architecture du système.

UML (Unified Modeling Language) est un langage de modélisation unifié permet de modéliser une application logicielle d'une façon standard dans le cadre de conception orienté objet .

Il permet de couvrir le cycle de vie d'un logiciel depuis la spécification des besoins jusqu'au codage en offrant plusieurs moyens de description et de modélisation des acteurs et de utilisation système, du comportement des objets, du flot de contrôle internes aux opérations, des composants d'implémentation et leurs relations, de la structure matérielle et de la distribution des objets et des composants

indépendamment des techniques d'implémentation et peut être mis à jour selon les besoins .

### **/ Avantages d'UML :**

- UML est un langage formel et normalisé : il permet un gain de précision et de stabilité.
- C'est est un support de communication performant : il permet grâce à sa représentation graphique, d'exprimer visuellement une solution objet, de faciliter la comparaison et l'évolution de solution.

### **/ Diagrammes d'UML utilisés :**

Depuis UML 2.3, il existe quatorze diagrammes autour desquels s'articule ce langage, chacun d'eux étant dédié à la représentation des concepts particuliers d'un système logiciel. Ces types de diagrammes sont répartis en trois grands groupes :

- Diagrammes de structure ou diagrammes statiques
- Diagrammes de comportement
- Diagrammes d'interaction ou diagrammes dynamiques

Durant notre étude, nous allons nous intéresser des diagrammes suivants :

✓ Diagramme de cas d'utilisation (avec l'ensemble de scénarios : nominal + cas d'erreur)

✓ Diagramme de séquence

Cette limitation est largement suffisante pour notre travail, à côté du modèle conceptuel de données pour compléter la tâche .

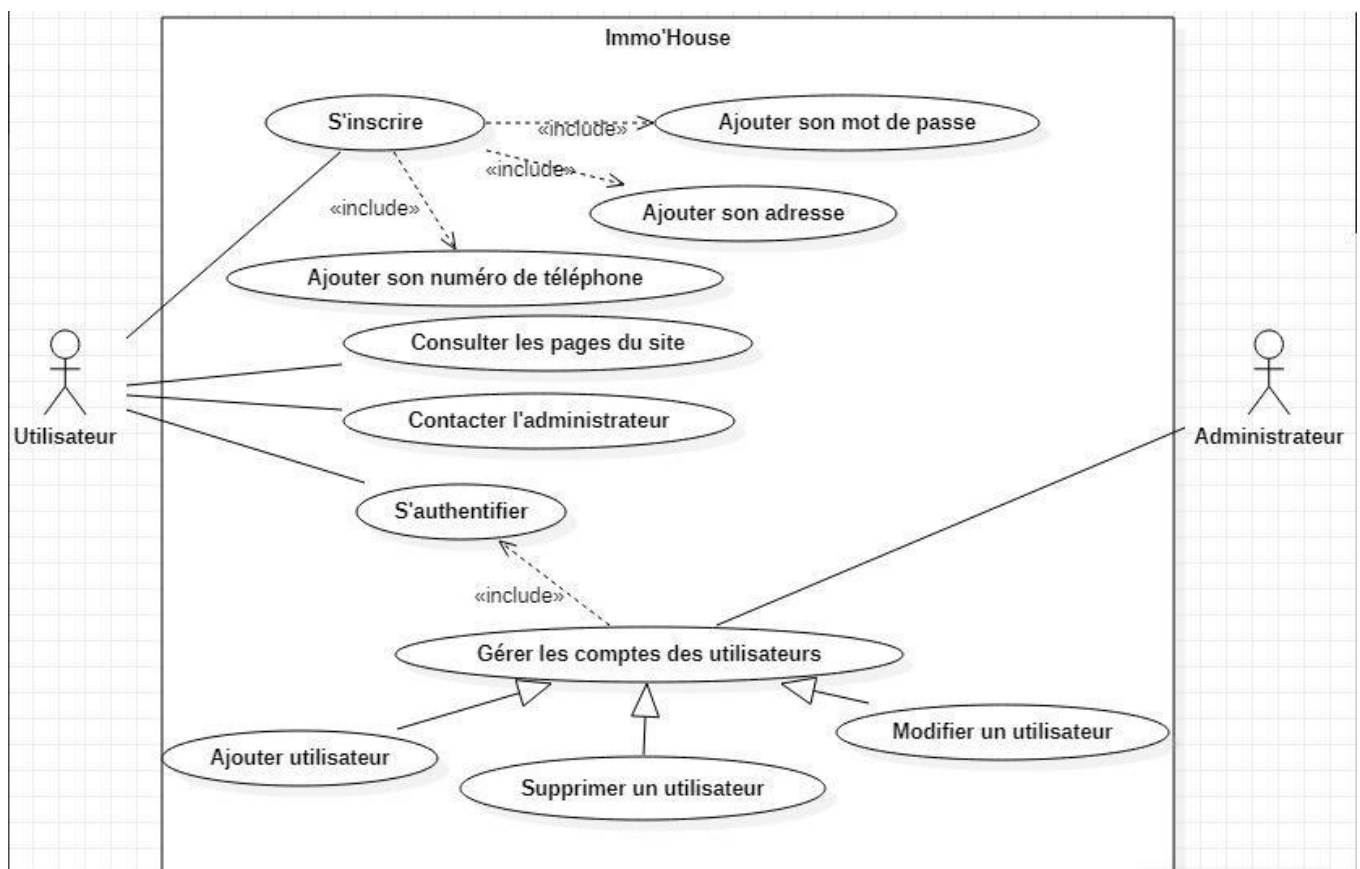
### 3. Conception détaillée :

#### / **Diagramme de cas d'utilisation général :**

**Le diagramme de cas d'utilisation a pour but de donner une vision globale sur les interfaces de future application.**

C'est le premier diagramme UML constitué d'un ensemble d'acteurs qui agit sur des cas d'utilisation et qui décrit sous la forme d'actions et des réactions, le comportement d'un système du point de vue utilisateur.

La figure suivante représente le diagramme de cas d'utilisation général lié à notre application :



**Figure: Diagramme de cas d'utilisations général**



## / Description détaillé des cas d'utilisations :

### a. Scénario (nominal+cas d'erreur) du cas d'utilisation "S'authentifier" :

scénario nominal	
Action de l'acteur	Réaction de système
1-L'utilisateur demande une connexion au système	2-Le système affiche l'interface de connexion
3-Il saisi son nom d'utilisateur et son mdp et appuie sur le bouton "Connexion"	4-Le système vérifie si l'utilisateur existe dans la base de donnée , et le redirige vers son profil
Dans le cas où l'utilisateur saisit un nom d'utilisateur ou un mot de passe erroné le système le redirige vers la page de connexion de nouveau, sinon s'il n'est pas encore inscrit il clique sur le bouton « pas encore inscrit » .	

**Tableau1: Scénario du cas d'utilisation "S'authentifier"**

**b. Scénario du cas d'utilisation "S'inscrire":**

Scénario nominale	
Action de l'acteur	Réaction de système
1-L'utilisateur appuie sur le bouton s'inscrire  3-L'utilisateur remplit les champs et appuie sur le bouton « s'inscrire »	2-Le système affiche l'interface d'inscription  4-Le système lui dirige vers la page d'accueil
Dans le cas où l'utilisateur saisit un nom d'utilisateur déjà existant ou un mot de passe erroné le système le redirige vers la première page pour choisir un autre nom d'utilisateur ou retaper le mot de passe correctement.	

**Tableau2: Scénario du cas d'utilisation "S'inscrire"**

# REALISATION

## **1. Introduction :**

Cette partie sera essentiellement consacrée à la présentation des principales interfaces du système sous forme de capture d'écrans. Chaque écran est précédé d'un commentaire pour décrire son fonctionnement.

Mais avant d'exposer les pages de notre petit site web, nous présentons aussi les outils de développement utilisés pour réussir la réalisation.

## **2. Outils de développement :**

### **/ HTML(HyperText Markup Language) :**

HTML est le langage de balisage conçu pour représenter les pages web .

### **/CSS (Cascading style sheets):**

Pour ajuster le design des pages de notre application , nous avons choisis CSS . Ce sont des feuilles de style qui permettent de gérer la présentation d'une page web . CSS permet de créer des règles de styles et de mise en pages. Ces règles portent sur le positionnement des éléments, l'alignement, les polices de caractères, les couleurs, les marges et espacement, les bordures, les images de fond, etc. Le but de CSS est de séparer la structure d'un document HTML et sa présentation.

### / JavaScript):

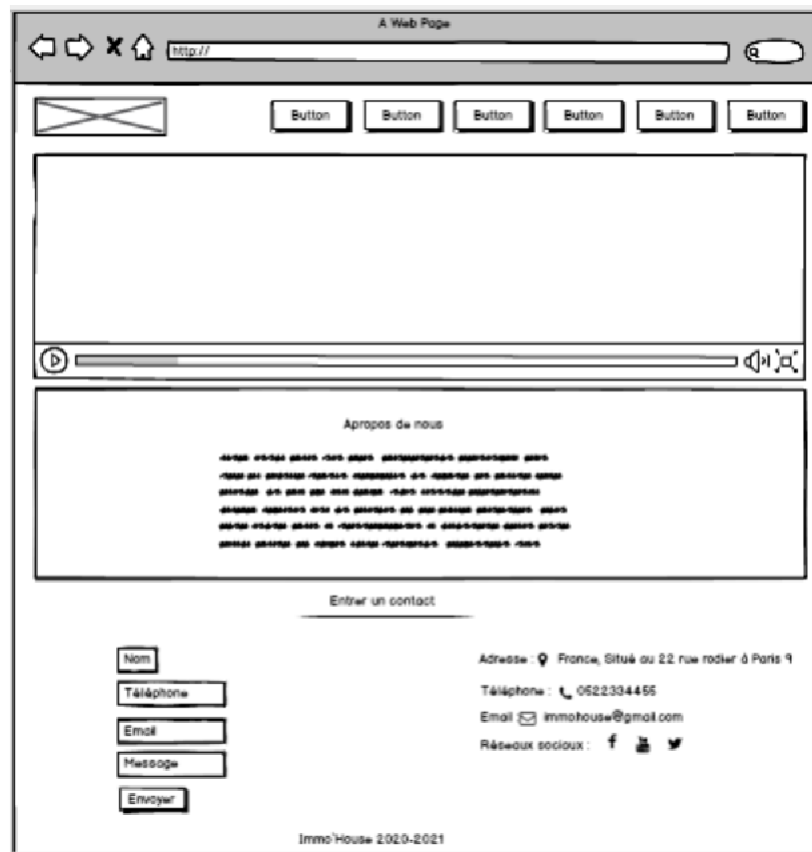
JavaScript est un langage de programmation de scripts principalement employé dans les pages web interactives et à ce titre est une partie essentielle des applications web. Avec les technologies HTML et CSS, JavaScript est parfois considéré comme l'une des technologies cœur du World Wide Web.

## 3. Interfaces de l'application :

### / Page d'accueil :

La page d'accueil contient un menu avec 6 boutons . Si l'utilisateur appuie sur l'un des 2 premiers boutons ou "contact" il sera dirigé vers une partie (dépend du bouton choisi) de la page d'accueil.

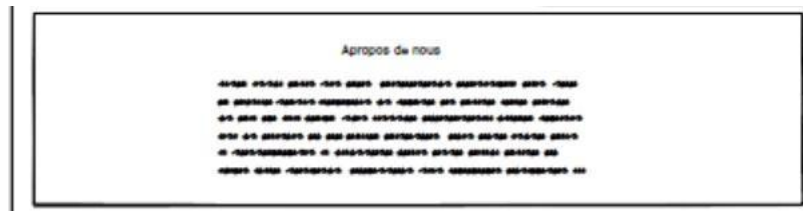
Si il appuie sur le bouton S'inscrire ou Se connecter il sera dirigé vers la page d'inscription ou de connexion selon les choix.



**Figure: Page d'accueil**



**Figure: Vidéo**



**Figure: Partie "à propos de nous"**

Entrer un contact

Nom	Adresse : France, Situé au 22 rue rodier à Paris 9
Téléphone	Téléphone : 0522334455
Email	Email : immohouse@gmail.com
Message	Réseaux sociaux : f y t
Envoyer	

Immo'House 2020-2021

**Figure: Partie "Contact"**