



UNIVERSITÉ DE LORRAINE

Filière : Ingénierie Logicielle (IL)

Module : Technologie Web

Cahier des charges du projet web UnToitPourTous

Projet académique

Étudiants :

- Kaouthar ASRAR
- Meriem AIT EL ACHKAR

Enseignant : Simon Hajek

27 septembre 2025

Table des matières

1	Présentation générale du projet	2
1.1	Contexte	2
2	Objectifs du projet	2
2.1	Objectif général	2
2.2	Objectifs spécifiques	2
3	Public cible	3
3.1	Description générale	3
3.2	Segments de public cible	3
3.3	Caractéristiques des utilisateurs	3
4	Fonctionnalités attendues	3
4.1	Description générale	3
4.2	Fonctionnalités principales	3
4.3	Fonctionnalités techniques complémentaires	4
5	Spécifications techniques	4
5.1	Architecture générale	4
5.2	Sécurité et authentification	4
5.3	Environnements et déploiement	4
5.4	Standards de développement	5
6	Design et ergonomie	5
7	Contraintes du projet	5
8	Livrables attendus	5
9	Tests et validation	5
9.1	Objectif	5
9.2	Types de tests	5

1 Présentation générale du projet

1.1 Contexte

Le marché immobilier français connaît une forte digitalisation. De plus en plus de personnes utilisent des plateformes en ligne pour rechercher, louer ou acheter un logement. Ces plateformes se concentrent principalement sur la diffusion d’annonces et la mise en relation simple entre propriétaires et candidats.

Cependant, les utilisateurs expriment des besoins nouveaux : plus de transparence, de fiabilité dans les informations fournies, et des outils qui facilitent la préparation des démarches (constitution du dossier locataire, échanges sécurisés, etc.).

Dans ce contexte, notre projet vise à développer une application web moderne de location et de vente de maisons en France, conçue pour simplifier et enrichir l’expérience des utilisateurs. Elle reprend les fonctions de base attendues d’une plateforme immobilière (consultation et publication d’annonces, gestion des profils, recherche et filtres), tout en intégrant des éléments différenciateurs qui apportent de la valeur ajoutée et renforcent la confiance entre les parties.

2 Objectifs du projet

2.1 Objectif général

Développer une application web moderne de location et de vente de maisons en France permettant de faciliter la mise en relation entre propriétaires et locataires / acheteurs, en offrant une expérience utilisateur fluide, sécurisée et transparente, et en respectant les standards professionnels du développement Full-Stack.

2.2 Objectifs spécifiques

- Concevoir une interface intuitive et responsive permettant aux utilisateurs de rechercher, filtrer et consulter facilement des annonces immobilières.
- Permettre aux propriétaires et agences de publier et gérer leurs annonces.
- Mettre en place un système d’authentification et de gestion des comptes utilisateurs (propriétaires, locataires, administrateurs) sécurisé par JWT.
- Gérer la base de données des biens immobiliers et des utilisateurs de façon structurée et évolutive (MongoDB + Mongoose).
- Faciliter et sécuriser les échanges entre utilisateurs via une messagerie interne et/ou anonymisée (atout différenciateur).
- Introduire des éléments de confiance et de transparence (comme la possibilité pour les anciens locataires de donner un retour d’expérience sur le logement) pour se différencier des plateformes existantes.
- Respecter les bonnes pratiques de développement : architecture modulaire côté backend (NestJS), code propre et commenté, gestion des erreurs.
- Préparer la documentation complète (installation, architecture, choix techniques) et un déploiement en ligne pour rendre l’application accessible et valoriser le projet.

3 Public cible

3.1 Description générale

L'application s'adresse à l'ensemble des acteurs du marché immobilier français cherchant à simplifier et sécuriser leurs démarches de location ou d'achat/vente de biens immobiliers.

3.2 Segments de public cible

- **Propriétaires et agences immobilières** : Publier, gérer et promouvoir leurs annonces via un tableau de bord intuitif.
- **Candidats locataires et acheteurs** : Rechercher et filtrer des biens, constituer un dossier numérique, échanger avec les propriétaires.
- **Administrateurs et modérateurs** : Superviser la plateforme, garantir la conformité des annonces et le respect des règles de confidentialité.

3.3 Caractéristiques des utilisateurs

Cette cible comprend principalement des utilisateurs connectés (PC et mobile), souhaitant un outil en ligne moderne et sécurisé, qui se distingue par sa transparence et son ergonomie.

4 Fonctionnalités attendues

4.1 Description générale

Cette section présente les principales fonctionnalités attendues de l'application web. Elles sont regroupées en plusieurs catégories afin de couvrir les besoins des différents types d'utilisateurs (propriétaires, locataires/acheteurs, administrateurs).

4.2 Fonctionnalités principales

- **Gestion des utilisateurs** : Création, authentification et gestion des comptes pour les propriétaires, les locataires/acheteurs et les administrateurs, avec un système sécurisé basé sur JWT.
- **Gestion des annonces immobilières** : Publication, modification, suppression et consultation d'annonces de location et de vente de biens immobiliers.
- **Recherche et filtrage** : Possibilité pour les utilisateurs de rechercher des biens selon différents critères (prix, localisation, type de bien, surface, nombre de pièces...).
- **Messagerie interne** : Mise en place d'un système d'échanges sécurisé entre utilisateurs via une messagerie interne ou anonymisée.
- **Tableau de bord utilisateur** : Accès à un espace personnel permettant de gérer ses annonces, ses messages et ses informations de profil.
- **Éléments de transparence et de confiance** : Possibilité pour les anciens locataires de donner un retour d'expérience sur le logement afin d'instaurer plus de confiance et de se différencier des plateformes existantes.

- **Dossier locataire numérique** : Mise à disposition d'un espace permettant aux locataires/acheteurs de constituer un dossier numérique sécurisé et partageable avec les propriétaires.

4.3 Fonctionnalités techniques complémentaires

- **Sécurité et RGPD** : Respect des bonnes pratiques de développement, gestion des erreurs, protection des données personnelles et conformité au RGPD.
- **Documentation et déploiement** : Préparation d'une documentation technique complète (installation, architecture, choix techniques) et mise en ligne de l'application (bonus).

5 Spécifications techniques

5.1 Architecture générale

L'application sera conçue en architecture *Full-Stack* avec une séparation claire entre le **frontend** et le **backend**. Le frontend communiquera avec le backend via une API REST sécurisée, et le backend interagira avec une base de données NoSQL.

- **Frontend** : Développé en React, avec hooks et React Router pour le routage. Gestion d'état cohérente via Context API ou Redux.
- **Backend** : Développé en NestJS avec architecture modulaire, services, contrôleurs et validation des données.
- **Base de données** : MongoDB avec Mongoose pour la gestion des schémas et des relations.

5.2 Sécurité et authentification

L'application doit assurer la sécurité des données et des échanges utilisateurs :

- **Authentification et autorisation** : Utilisation de JWT pour protéger les routes sensibles.
- **Gestion des droits** : Différenciation des permissions selon les rôles (propriétaires, locataires/acheteurs, administrateurs).
- **Protection des données personnelles** : Conformité au RGPD (chiffrement, anonymisation si nécessaire).

5.3 Environnements et déploiement

Le projet doit être configuré pour fonctionner sur différents environnements (développement, test, production) :

- **Variables d'environnement** : Utilisation de fichiers .env pour les clés et paramètres sensibles.
- **Déploiement** : Hébergement sur une plateforme cloud (Heroku, Vercel, Render ou équivalent) pour rendre l'application accessible en ligne.
- **Documentation technique** : Fourniture d'un README détaillé et d'une documentation d'installation.

5.4 Standards de développement

Le code doit respecter les standards de qualité professionnelle :

- **Clean code** : Respect des principes DRY et SOLID, code commenté et structuré.
- **Gestion des erreurs** : Mise en place de middlewares pour la gestion globale des erreurs dans le backend.
- **Versioning** : Utilisation cohérente de Git avec des commits explicites.

6 Design et ergonomie

7 Contraintes du projet

- **Projets parallèles** : L'équipe travaille sur plusieurs projets issus d'autres modules en parallèle, ce qui limite le temps disponible pour ce projet.
- **Délais de livraison** : La présentation du projet est prévue la dernière séance de cours, à 20h dans la salle de classe.
- **Planification serrée** : Les différentes étapes (analyse, conception, développement, tests) doivent être planifiées pour respecter ce délai.
- **Ressources limitées** : Les environnements de développement et de déploiement devront être configurés avec des outils gratuits ou académiques.

8 Livrables attendus

- Code source complet
- Repository Git
- Documentation technique et fonctionnelle
- Présentation finale
 - Démonstration fonctionnelle de l'application.
 - Support visuel (slides) expliquant le concept et la mise en œuvre.
- Déploiement en ligne (bonus)

9 Tests et validation

9.1 Objectif

Vérifier que l'application respecte les spécifications fonctionnelles et techniques et qu'elle fonctionne de manière fiable et sécurisée.

9.2 Types de tests

- **Unitaires** : Vérifier chaque composant ou service.
- **Intégration** : Vérifier l'interaction frontend-backend-base de données.
- **Fonctionnels** : Tester les principales fonctionnalités (authentification, annonces, messagerie...).
- **Sécurité** : Contrôler la protection des données et des accès.