

Cahier des charges

Plateforme de mise en relation entre entreprises et investisseurs selon leur localisation

Table of Contents

Introduction	2
Présentation du projet.....	2
1. Contexte du projet.....	2
2. Objet du projet.....	2
3. Organisation du document.....	3
Besoins fonctionnels de la nouvelle application	3
1. Nouvelle interface utilisateur	3
2. Interface web de front-office	3
3. Interface web de back-office.....	4
4. Interface web pour les membres authentifiés.....	4
5. Interface Applicative de Programmation (API)	4
Besoins techniques de la nouvelle application	5
1. La plateforme, le langage, le modèle et le framework	5
2. La base de données.....	6
3. Interface web.....	6
4. Système d'authentification	6
5. Performance	7
Conception	7
1. Diagramme de classe	7
2. Diagramme de classe	8
Planification et livrables	8
Conclusion.....	9

Introduction

Le projet **Real Estate Invest** vise à démocratiser l'investissement immobilier grâce à une plateforme numérique moderne et accessible, inspirée de modèles comme GetStake. Il permet aux utilisateurs d'investir de manière fractionnée dans des biens immobiliers et de générer des revenus passifs.

Porté par une volonté d'inclusion, de sécurité et de transparence, le projet s'adresse aussi bien aux particuliers qu'aux professionnels. La plateforme comprendra des services de gestion des biens, des utilisateurs, des investissements et des transactions, reposant sur une infrastructure cloud performante.

Ce cahier des charges définit les spécifications fonctionnelles et techniques de la solution, les besoins utilisateurs, les fonctionnalités attendues, et les contraintes du projet.

Présentation du projet

1. Contexte du projet

Le projet **Real Estate Invest** vise à développer une plateforme web d'investissement immobilier fractionné, inspirée de GetStake. Aujourd'hui, l'accès à ce type d'investissement reste limité, peu intuitif ou réservé à une clientèle experte.

Cette nouvelle application offrira une interface moderne, sécurisée et accessible à tous, permettant d'acheter des parts de biens immobiliers, de suivre les performances et de gérer un portefeuille en ligne. Le système comprendra une interface utilisateur responsive, un espace d'administration, une base de données optimisée et l'intégration d'un prestataire de paiement.

Ce projet répond aux nouvelles attentes en matière d'investissement digital : simplicité, transparence et accessibilité.

2. Objet du projet

Le projet **Real Estate Invest** a pour objectif la création d'une plateforme web dédiée à l'investissement immobilier fractionné. Pour cela, les développements suivants sont attendus :

- Mise en place d'une base de données optimisée, adaptée aux besoins fonctionnels du projet ;
- Développement d'une interface web complète (front-office et back-office) intégrant :
 - un design moderne pour la consultation des biens immobiliers,
 - un espace utilisateur sécurisé pour la gestion des investissements,
 - un tableau de bord d'administration pour la gestion des biens, des utilisateurs et des transactions ;
- Intégration d'un prestataire de paiement pour la gestion des opérations financières ;

- Déploiement d'une API REST pour permettre la communication avec des systèmes tiers (mobile, outils analytiques, etc.).

3. Organisation du document

Le présent document est structuré de la manière suivante :

- Présentation des besoins fonctionnels de la plateforme Real Estate Invest,
- Description des besoins techniques et des contraintes liées à l'architecture de la solution,
- Planification du projet et liste des livrables attendus à chaque étape,

Besoins fonctionnels de la nouvelle application

La plateforme Real Estate Invest devra proposer une solution complète pour permettre à tout utilisateur d'investir facilement, en toute sécurité, dans des biens immobiliers fractionnés. Elle s'articulera autour de plusieurs composants fonctionnels clés :

1. Nouvelle interface utilisateur

La nouvelle interface devra être :

Intuitive	une expérience utilisateur fluide et claire,
Moderne	un design attractif et professionnel, inspiré des meilleures pratiques UX/UI actuelles,
Multilingue	au minimum une version française et une version anglaise

2. Interface web de front-office

L'interface publique permettra :

- La **consultation des projets immobiliers disponibles** avec leurs détails (localisation, description, photos, rendements attendus, durée, etc.),
- La **visualisation des statistiques agrégées** (projets en cours, montants collectés, rendement global, etc.),
- Le **processus d'investissement**, incluant :
 - La sélection de parts,
 - La simulation de gains,
 - Le paiement en ligne via un prestataire sécurisé,
 - La génération automatique de documents légaux (contrat d'investissement, etc.),

3. Interface web de back-office

L'administration de la plateforme devra disposer d'un espace sécurisé avec les fonctionnalités suivantes :

- **Gestion des utilisateurs** (création, suspension, modification, rôles, vérification KYC),
- **Gestion des projets immobiliers** (ajout, mise à jour, publication/dépublication),
- **Gestion des investissements** (suivi des montants investis, affectation des parts, paiements),
- **Suivi des flux financiers** (rapports de revenus générés, commissions, paiements automatiques ou manuels),
- **Gestion du contenu** visible sur le front-office (textes, images, FAQ, articles de blog, etc.),
- **Outils analytiques et tableaux de bord** pour suivre l'activité, les performances et les comportements utilisateurs.

4. Interface web pour les membres authentifiés

Une fois connectés, les utilisateurs accèdent à un espace personnel sécurisé leur permettant de :

- Suivre **leur portefeuille d'investissements**,
- Visualiser **leurs revenus passifs générés**,
- Télécharger leurs **documents légaux et fiscaux**,
- Recevoir des **notifications** sur les projets ou les mises à jour,
- Mettre à jour leurs **informations personnelles**,
- Réaliser de nouveaux **investissements** rapidement.

5. Interface Applicative de Programmation (API)

Une API RESTful devra être développée pour :

- **Permettre l'interopérabilité** avec des applications mobiles ou d'autres interfaces web,
- **Exposer des endpoints sécurisés** pour accéder aux données publiques (projets immobiliers) et privées (portefeuille utilisateur, transactions),

- **Faciliter l'automatisation de certaines tâches** comme l'envoi de rapports, l'agrégation des statistiques, ou la synchronisation avec des systèmes partenaires (fournisseurs de paiement, CRM, etc.),

Besoins techniques de la nouvelle application

La mise en place de la plateforme **Real Estate Invest** repose sur une série de choix techniques visant à garantir la performance, la sécurité, et la scalabilité de la solution. Les principales spécifications techniques sont les suivantes :

1. La plateforme, le langage, le modèle et le framework

La plateforme devra être développée sur des technologies robustes et largement utilisées afin de garantir sa pérennité et sa compatibilité avec les standards industriels :

- **Plateforme** : Cloud (préférentiellement AWS) pour une scalabilité optimale et une gestion centralisée des ressources,
- **Langages de programmation** :

Frontend	React.js pour garantir une interface utilisateur moderne, réactive et performante,
Backend	Spring boot (Java) pour sa capacité à gérer des requêtes concurrentes de manière performante et évolutive,

- **Modèle et architecture** : Architecture **microservices** ou **monolithique modulaire**, selon l'analyse des besoins, permettant une gestion souple des différentes fonctionnalités de la plateforme,
- **Frameworks**:

Frontend	Utilisation de frameworks CSS modernes (TailwindCSS) pour garantir une expérience utilisateur fluide,
Backend	Utilisation de frameworks robustes pour la gestion des API (Spring Boot pour Java).

2. La base de données

La base de données devra répondre aux exigences de volume et de complexité des données immobilières et financières. Les spécifications techniques sont :

- **Type de base de données :**
 - **Relationnelle** (MySQL) pour la gestion des utilisateurs, investissements, transactions et historiques,
 - **NoSQL** (MongoDB) pour le stockage des données non structurées ou de session (logs, statistiques).
- **Sécurité :**
 - Chiffrement des données sensibles,
 - Gestion des accès basée sur des rôles avec permissions détaillées.

3. Interface web

L'interface web devra être conçue de manière à répondre aux exigences de performance et de compatibilité avec les standards actuels :

- **Accessibilité** : La plateforme devra respecter les standards d'accessibilité (WCAG 2.1) afin de garantir une expérience utilisateur optimale pour tous,
- **Optimisation des performances** : Chargement rapide des pages grâce à l'utilisation de techniques modernes (Lazy loading, optimisation des images, réduction des appels API, etc.),
- **Compatibilité** : L'application doit être compatible avec les principaux navigateurs (Chrome, Firefox, Safari, Edge).

4. Système d'authentification

Le système d'authentification est essentiel pour assurer la sécurité des utilisateurs et la confidentialité des données. Il inclura :

- **Authentification via email et mot de passe** avec validation des données de l'utilisateur,
- **Gestion des rôles** : Différenciation des droits d'accès selon les utilisateurs (administrateurs, investisseurs),
- **Authentification via des services tiers** comme Google, Facebook, ou des solutions d'authentification SSO pour faciliter l'accès des utilisateurs.

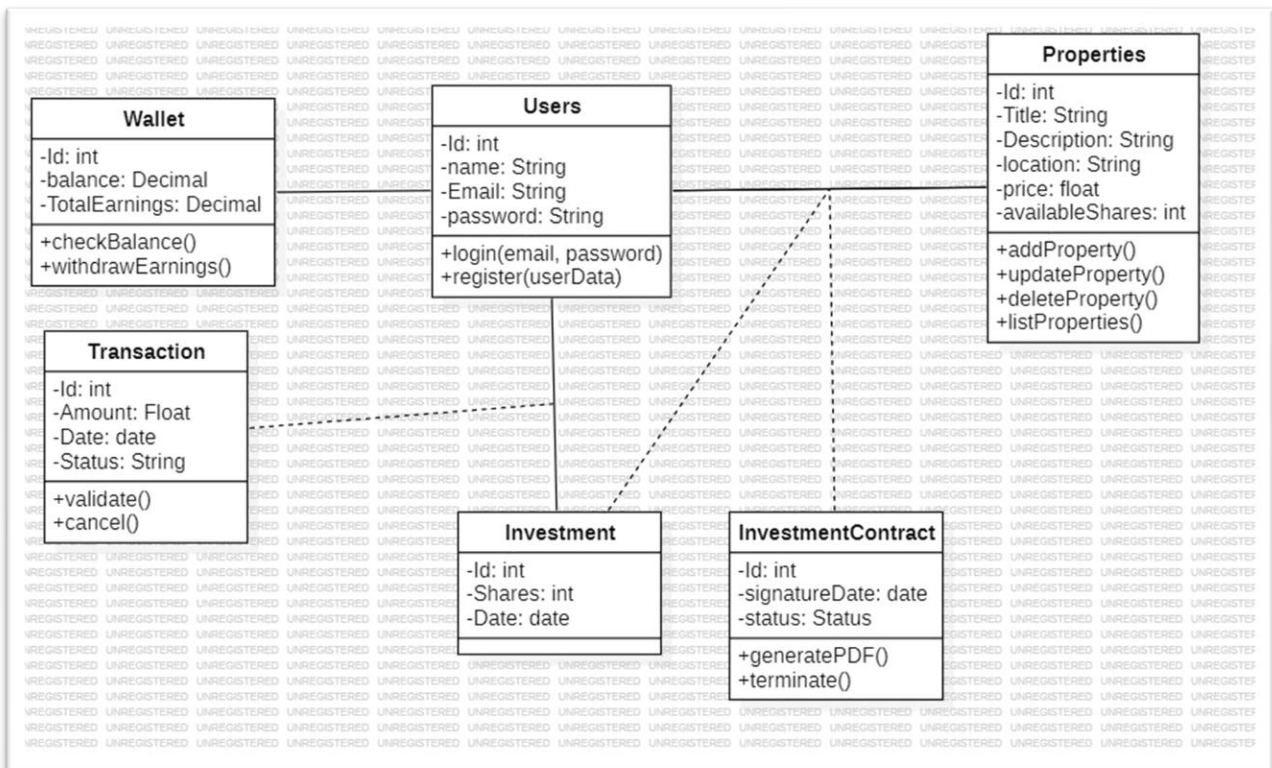
5. Performance

La plateforme doit être capable de gérer un grand volume d'utilisateurs et d'investissements, tout en offrant une expérience fluide et rapide. Les exigences en matière de performance incluent :

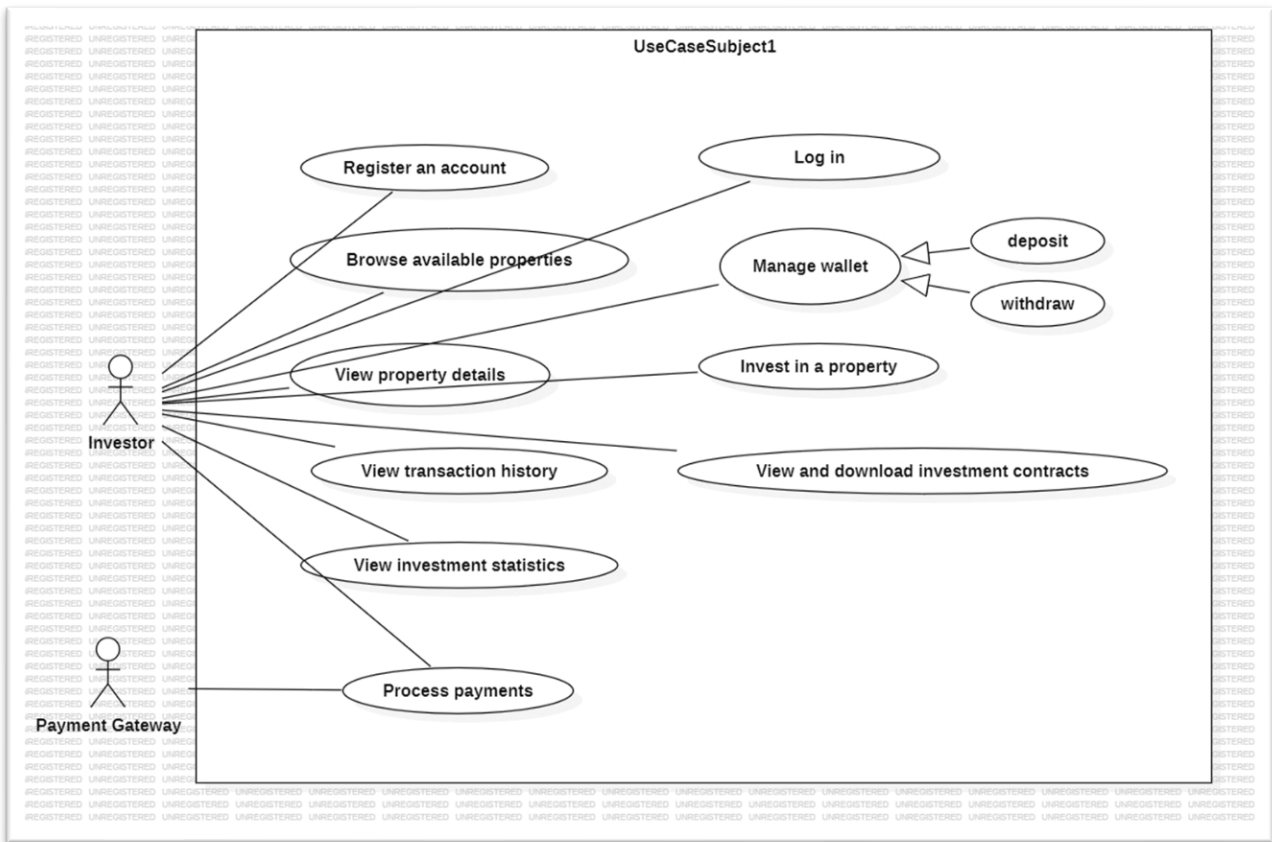
- **Scalabilité horizontale** : L'architecture de l'application doit permettre d'ajouter facilement des ressources supplémentaires en cas d'augmentation du trafic utilisateur,
- **Latence minimale** : Réduction des délais de réponse grâce à une architecture optimisée et un équilibrage de charge efficace,
- **Haute disponibilité** : Utilisation d'une infrastructure redondante pour garantir une disponibilité continue de la plateforme, même en cas de défaillance d'un service,
- **Tests de charge** : Mise en place de tests de performance et de montée en charge pour identifier et résoudre les goulots d'étranglement avant le déploiement en production.

Conception

1. Diagramme de classe



2. Diagramme de Cas d'utilisation



Planification et livrables

Les livrables principaux à fournir à chaque étape du projet sont les suivants :

1. **Cahier des charges validé** : Document finalisé avec la description complète des besoins fonctionnels et techniques, validé par toutes les parties prenantes,
2. **Code source du backend et frontend** : Code bien documenté et versionné, prêt pour la mise en production,
3. **Rapports de tests** : Rapports détaillant les tests effectués (fonctionnels, de sécurité, de performance) et les corrections apportées,
4. **Application en production** : L'application entièrement fonctionnelle et déployée sur l'infrastructure choisie,

Conclusion

Le projet **Real Estate Invest** vise à démocratiser l'investissement immobilier grâce à une plateforme numérique moderne et accessible. Ce cahier des charges définit les besoins fonctionnels et techniques pour développer une application robuste, performante et sécurisée, offrant une expérience fluide pour les utilisateurs, qu'ils soient particuliers ou professionnels. En suivant les étapes de développement et de planification, ce projet répondra aux objectifs stratégiques et apportera une réelle valeur ajoutée à l'investissement immobilier numérique.