République Tunisienne
*****
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
*****
Université de Monastir
*****
Institut Supérieur d'Informatique et de Mathématiques de Monastir
*****
Département Informatique

N° d'ordre : LINFO**

Project Memory
End of Studies

Presented with a view to obtaining the

National Bachelor of Science Diploma

Computer Science

Speciality :

Software and Information System Engineering

By

Jaziri Ahmed

Development of a mobile application and web back office dedicated to real estate investment

Defended on *** in front of the jury composed of:

Mr :	**President**
Mr :	**Reporter**
Ms. : Bali Nadia	**Educational Supervisor**
Mr. : ZOUARI Khalil	**Technical Supervisor**

SUMMARY

This work is part of the completion of our Final Year Project at the Higher Institute of Computer Science and Mathematics of Monastir for the academic year 2024-2025, aiming for the National Fundamental License Diploma in Computer Science. Conducted within the company **'KZ IT Services'**, our main objective is the development of a mobile application and a web back-office dedicated to real estate investment named **'KORPOR'**. We used MySQL to manage the databases, Express-Node.js to implement the backend, and React and Vite to implement the frontend. Project management followed the SCRUM Agile methodology, emphasizing agile practices such as sprint planning, sprint management, and regular meetings.

Keywords: MySQL, Express-Node.js, React, Vite.

ABSTRACT

This project is part of our graduation project at the Higher Institute of Computer Science and Mathematics of Monastir for the 2024-2025 academic year, leading to the national diploma of fundamental license in Computer Science. Carried out within the company **"KZ IT Services"**, our main objective is the development of a mobile application and web back office dedicated to real estate investment called **"KORPOR"**. We used MySQL to manage the databases, Express-Node.js for the backend implementation, and React and Vite for the frontend. The project management followed the SCRUM Agile methodology, with an emphasis on agile practices such as sprint planning, sprint management, and regular meetings.

Keywords: MySQL, Express-Node.js, React, Vite.

Dedication

Acknowledgement

TABLE OF CONTENTS

Contents

GENERAL INTRODUCTION

"Innovation distinguishes between a leader and a follower."

— Steve Jobs

In today's rapidly evolving financial landscape, traditional investment methods are often burdened by opaque processes, cumbersome bureaucracy, and significant entry barriers. Investors have long struggled with outdated systems that impede transparency, elevate risks, and complicate access to promising opportunities. Such challenges not only limit diversification but also expose users to uncertainties that modern technology can easily overcome.

Korpor was conceived to transform this paradigm by delivering a fully integrated, AI and blockchain-powered mobile investment platform. By harnessing advanced data analytics, machine learning, and cutting-edge blockchain technology, Korpor streamlines every facet of the investment process. The application offers a seamless user onboarding experience, intuitive project listings enriched with AI-driven recommendations, and a secure, automated investment flow that simplifies transactions while ensuring that every operation is recorded immutably on the blockchain. Investors can manage their portfolios effortlessly through a comprehensive dashboard, with real-time notifications, an interactive AI chatbot, and multi-language support delivering a personalized and globally accessible experience.

Security and trust are at the heart of Korpor's design. By employing state-of-the-art encryption, blockchain-based transparency, and strict compliance measures, the platform safeguards sensitive financial data and guarantees that every transaction is executed within a secure and verifiable framework. Continuous monitoring, performance optimization, and the immutable nature of blockchain records further ensure that the application remains resilient, scalable, and resistant to fraud in a dynamic market environment.

Developed under a flexible Agile framework that combines iterative development with strategic project management best practices, Korpor is designed to rapidly adapt to evolving market trends and user needs. This methodical approach allows for regular feedback, swift enhancements, and the seamless integration of innovative features throughout the development lifecycle.

Document Structure

- The first chapter, **Project Context**, delves into the industry challenges and the vision that inspired Korpor's creation.
- The second chapter, **Preparation Phase**, outlines the comprehensive requirements gathering, needs analysis, architectural design, and the selection of cutting-edge tools and technologies.
- Subsequent chapters document the progressive implementation of core features—from AI-enhanced project recommendations and blockchain-secured transactions to comprehensive portfolio management—each developed through clearly defined sprints encompassing analysis, design, and deployment phases.

Through this structured approach, we demonstrate how Korpor leverages modern technology to reimagine investment management, offering a secure, transparent, and dynamic solution that is set to redefine digital financial engagement.

CHAPTER 1

Project Context

1.1 Introduction

The aim of this chapter is to present the general framework of the Korpor project, a solution dedicated to real estate investment. In this section, we'll discuss successively:

- The presentation of the host organization.
- The context and challenges of the real estate sector.
- Analysis of existing solutions and identification of their limitations.
- Definition of functional and non-functional needs of the mobile application and web back office.

1.2 Project Context

This work is part of the end-of-study project for the national diploma of Applied Bachelor's degree in Computer Science from the Higher Institute of Computer Science and Mathematics of Monastir (ISIMM) for the year 2024/2025. I had the opportunity to do my end-of-study internship at the company "KZ IT Services", under the supervision of Mr. Khalil Zouari.

1.3 Hosting Company

The purpose of this section is to present the company within which I developed my project.

1.3.1 Hosting Company

KZ IT Service is a dynamic software company dedicated to delivering innovative IT solutions tailored to modern business needs. We specialize in designing and developing robust,

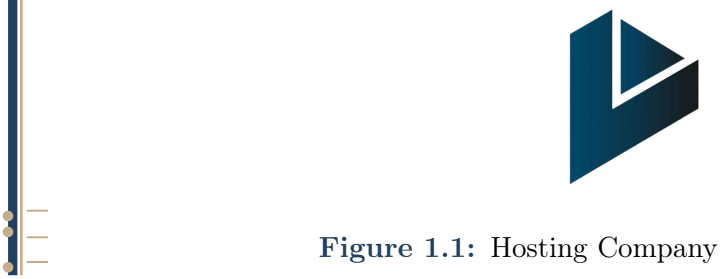

Figure 1.1: Hosting Company "KZ IT Services"

scalable applications that drive efficiency and digital transformation. Our experienced team leverages cutting-edge technology to create customized software that exceeds client expectations. With a strong commitment to quality and continuous improvement, we build lasting partnerships based on trust and excellence. At "KZ IT Service", innovation is at the core of everything we do, empowering our clients to achieve sustainable growth and success.

1.4 Preliminary Study

This preliminary study provides a review of some existing investment and asset management platforms. Further, the next section identifies some key concepts that will lead to further understanding of the domain in question.

1.4.1 Existing Solutions Study

For understanding the present scenario and to clearly demarcate our goals, some renowned investment platform analyses offering similar features, including "Aseel" and "Stake", are performed.

1.4.2 Available solutions and analysis

The Aseel Platform

Aseel is a portal through which users can invest in different real estate projects with ease. The interface allows the clients to surf various investment opportunities, view the details of the properties, and then make an informed decision. Aseel introduces transparency in the investment process by offering financial data, updates regarding projects, and returns that are estimated. This platform comes with an easy-to-use dashboard through which one tracks their investments and manages their assets without any hassle.

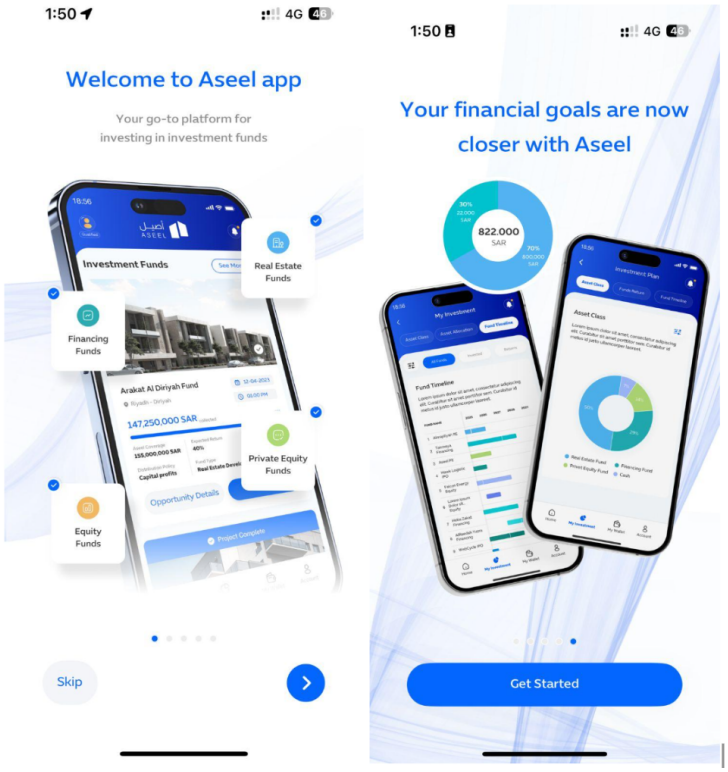

Figure 1.2: Interface of "The Aseel Platform"

The Stake Platform

Stake is an online investment platform that deals with real estate crowdfunding. It provides the opportunity to invest in fractions of property ownership, hence diversifying a portfolio without huge capital. On Stake, there are AI-powered recommendations based on user preferences, seamless payment integration, and a secure environment for investment. Besides, liquidity is guaranteed by enabling exit options for investors who may want to sell their shares in ongoing projects.

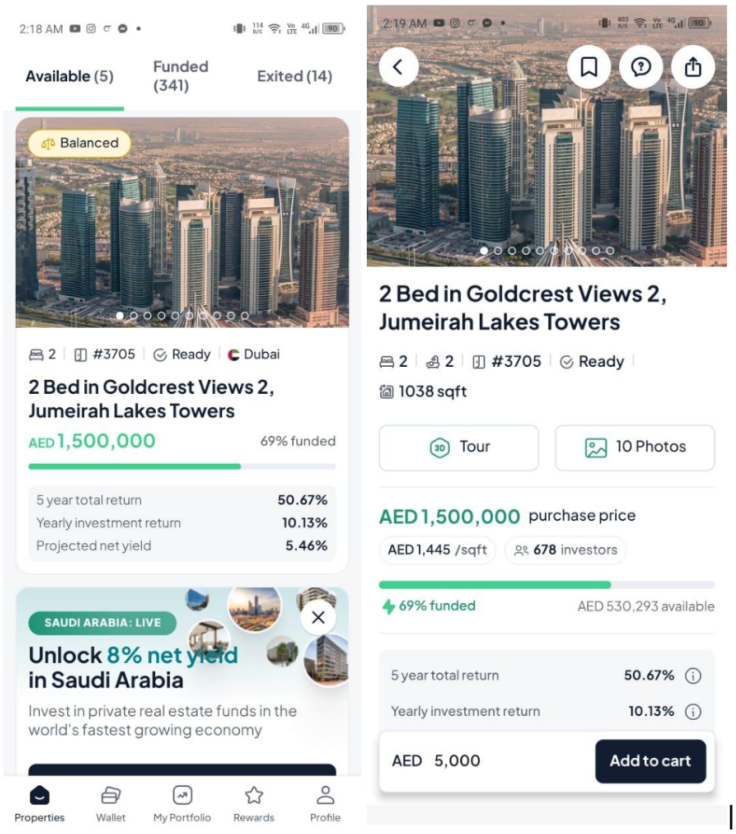

Figure 1.3: Interface of "The Stake Platform"

1.4.3 Comparative and Critical Analysis

We can summarize all that comes from our analysis based on a number of criteria used for the evaluation of these applications.

- **Speed (C1)**: The platform should obtain value for the user as fast as possible and effectively, anticipating their proliferating expectations.
- **Costs (C2)**: With minimum software development costs, it is important to keep the pricing predictable and acceptable.
- **Quality (C3)**: Since the market expects quality, any kind of error might affect brand reputation. Improvement of the platform should be regular.
- **Reliability (C4)**: Since modern-day investment platforms need to make sure of minimum downtime and maximum availability of services, this factor is critical.
- **Security (C5)**: Such an investment platform enforces access rights, roles, and contribution rights through a powerful security system.
- **Performance (C6)**: Crucial features include AI-powered recommendations going through seamlessly, easy transaction tracking, and investment monitoring.

- **Stability (C7)**: The platform should have a proven track record, regular updates, and a large user base to ensure its longevity.
- **Resilience (C8)**: In order to prevent data loss and guarantee a smooth experience for investors, it must be able to restore lost functionalities should issues occur.
- **User Experience (C9)**: The interface should be intuitive and user-friendly, hence allowing investors to move with ease through it, thus making wiser decisions.

Table 1.1: Evaluation Table

Solution	C1	C2	C3	C4	C5	C6	C7	C8	C9
Stake	✓	✓	✓	✓	✓	×	✓	✓	✓
Aseel	✓	✓	✓	×	✓	×	✓	×	✓

1.4.4 Proposed Solution

Having studied the already working platforms for investments, we found strengths and weaknesses that could define what was required from the project. Our proposed solution will look at:

- Developing an efficient mobile application for investment management.
- Increasing the level of users' engagement with recommendations using the power of AI.
- Ensuring responsive and user-friendly interaction with the interface.
- Gaining the trust of investors by ensuring transparency and security in the investing platform.
- Enhancing the security of data and following all the financial regulations.

The **Korpor** platform will be offering the following features:

- A directory of investment opportunities with deep financial insights into those opportunities.
- AI-driven recommendations of investments as per users' preferences.
- Smooth funding and payout mechanisms.
- Real-time portfolio performance tracking on a single screen/dashboard.
- Forum for interactive discussions on strategy and market trends among its users.
- Referral and Rewards System: An engaging system for rewarding users via referral.

1.5 Development methodology

The completion of the project on its delivery date is the main problem of every software development team. One of the most common problems encountered in the production of software is insufficient technical specifications, poor time management in the face of the use of emerging technology, and sudden changes in needs. In order to avoid these critical issues, we follow an agile methodology for project management.

1.5.1 SCRUM

Scrum is an agile development approach that is used to create software using incremental and iterative methods. Scrum is a quick, flexible, and efficient agile methodology that is intended to provide value to the client at every stage of the project's development. Scrum's main goal is to meet customer needs by fostering an atmosphere of open communication, group accountability, and constant improvement. The development process begins with a broad concept of what must be constructed, developing a list of features that the product owner desires, and arranging them according to priority (product backlog).

1.5.2 Agile Scrum roles and responsibilities

The Product Owner

Understands the customer and business requirements, then creates and manages the product backlog based on those requirements.

Responsibilities:

- Managing the scrum backlog
- Release management
- Stakeholder management

Developers

In Scrum, the term developer or team member refers to anyone who plays a role in the development and support of the product and can include researchers, architects, designers, programmers, etc.

Responsibilities:

- Delivering the work through the sprint
- To ensure transparency during the sprint, they meet daily at the daily scrum

Scrum Master

The role responsible for gluing everything together and ensuring that scrum is being done well. In practical terms, that means they help the product owner define value, the development team deliver the value, and the scrum team get better.

The Scrum Master focuses on:

- Transparency
- Empiricism
- Self-organization
- The Scrum events

1.5.3 The Scrum Events

The Scrum events are key elements of the Scrum Framework. They provide regular opportunities for enacting the Scrum pillars of Inspection, Adaptation and Transparency. In addition, they help teams keep aligned with the Sprint and Product Goals, improve Developer productivity, and remove impediments and reduce the need to schedule too many additional meetings.

- **Sprint**: All work in Scrum is done in a series of short projects called Sprints. This enables rapid feedback loops.
- **Sprint Planning**: The Sprint starts with a planning session in which the Developers plan the work they intend to do in the Sprint. This plan creates a shared understanding and alignment among the team.
- **Daily Scrum**: The Developers meet daily to inspect their progress toward the Sprint Goal, discuss any challenges they've run into, and tweak their plan for the next day as needed.
- **Sprint Review**: At the end of the Sprint, the Scrum Team meets with stakeholders to show what they have accomplished and get feedback.
- **Sprint Retrospective**: Finally, the Scrum Team gets together to discuss how the Sprint went and if there are things they could do differently and improve in the next Sprint.

1.6 Conclusion

In conclusion of this chapter, it is clear that planning and methodology are essential pillars to ensure the success of the project. By fully understanding the project framework, including the host organization's expectations and the challenges ahead, the team is better prepared to meet the challenges ahead.

This chapter lays the solid foundation on which the entire project will be built, providing a valuable guide for the next steps. The next chapter will allow us to analyze and specify the needs developed in our project.

CHAPTER 2

Analysis and Specification of Needs

2.1 Introduction

In this chapter, we will present the analysis and specification of needs. We start by presenting the specification of the requirements, illustrating them using the diagram of the global use cases. Then we will present our project architecture and our working environment, and finally we will present our product backlog and releases planning, and we will close our chapter with a conclusion.

2.2 Requirements Specification

In this section, we will define the actors of our application and the functional and non-functional needs that our application aims to fulfill.

2.2.1 Identifying Actors

We define actors as a shorthand for the roles played by entities outside the system that interact directly with them. In our system, we identify four types of actors:

- **Super Admin**: Responsible for the global configuration of the platform, they have extended privileges to manage administrators, oversee security, and ensure compliance. They can also configure advanced features and control all system resources.
- **Admin**: In charge of the day-to-day management of the platform, they can add, modify, or delete listings, supervise agency and user profiles, and ensure smooth operations. They are also responsible for monitoring and assisting other actors.
- **Real Estate Agent**: Dedicated to creating and updating real estate listings, they manage property information, handle investor requests, and finalize transactions

related to sales or rentals. They can also coordinate property visits and propose tailored offers.

- **Investor**: A user who wishes to browse and finance real estate projects. They have access to all available offers, can make investments in a few simple steps, and monitor the evolution of their portfolio. They also benefit from personalized insights to optimize their investments.
- **System**: The entity that automatically manages all basic functionalities, such as authentication, notification generation, transaction validation, and adherence to security protocols. It ensures the coherence and reliability of the application at all times.

2.2.2 Functional Requirements

After several meetings with our client, the various functional requirements of our application are illustrated as follows:

For the Super Admin (Korpor)

- **Authenticate**: The super admin enters their credentials to access the advanced management console.
- **Log Out**: After viewing or updating global settings, they can securely log out.
- **Manage Admin Accounts**: Create, enable/disable, or modify admin profiles associated with different real estate companies.
- **Monitor Security & Compliance**: Oversee transactions, data integrity, and regulatory adherence using specialized reporting and audit tools.
- **Configure Platform Features**: Define key parameters (payment methods, AI/blockchain integrations, etc.) and roll out feature updates.
- **View Global Reports**: Generate and analyze consolidated metrics (financials, user activity, transactions) for overall performance insights.
- **Moderate Content**: Review and remove any inappropriate or erroneous property listings or user-generated data.

For the Admin (Real Estate Company)

- **Authenticate**: The admin logs in with valid credentials to manage daily operations.
- **Log Out**: They can end their session to maintain account security.

- **Manage Real Estate Listings**: Add, update, or delete property listings visible to investors.
- **Oversee Real Estate Agents**: Create and manage agent accounts, assign properties, and monitor performance and commissions.
- **Track Transactions & Commissions**: Review incoming payments, calculate commissions owed to agents, and track the history of completed deals.
- **Address Investor Inquiries**: Respond to questions or concerns from investors, ensuring a smooth user experience.
- **Access Agency Dashboard**: View comprehensive statistics on properties, sales, rentals, and market trends.

For the Real Estate Agent

- **Authenticate**: The agent logs in to manage assigned properties and interact with potential investors.
- **Log Out**: Securely exit the account after completing tasks.
- **Manage Assigned Properties**: Create new listings, update property details, set prices, and upload images.
- **Handle Investment Requests**: Review purchase or rental offers, negotiate terms, and initiate contract finalization.
- **Contribute to AI Estimates**: Provide or refine data to improve AI-driven pricing and market analysis.
- **Maintain Client Relationships**: Communicate with investors, schedule property visits, and follow up on inquiries.
- **View Commissions**: Track earnings based on successful sales or rentals.

For the Investor (Mobile App User)

- **Create an account & authenticate**: Register to gain access to the platform's core features.
- **Log Out**: End the session to protect personal and financial data.
- **Browse Listings & Invest**: Explore available properties, filter according to preferences, and commit to an investment in a few steps.

- **Track Portfolio**: Monitor owned assets, property status, and receive real-time updates on performance.
- **Make Payments**: Use integrated payment methods (credit cards, digital wallets, etc.) to complete transactions.
- **Access AI Recommendations**: View data-driven insights and return-on-investment estimates generated by the system.
- **Manage Withdrawals & Earnings**: Withdraw profits, monitor rental income, or exit investments under the right conditions.

For the System

- **Automate Authentication**: Validate credentials, manage sessions, and maintain user roles and permissions.
- **Generate Notifications**: Send real-time alerts (e.g., new listings, completed transactions, commission updates) to relevant users.
- **Ensure Compliance & Security**: Leverage blockchain for data integrity, verify payments, and detect anomalies or fraudulent activities.
- **Coordinate AI Insights**: Aggregate and analyze real estate data to produce market predictions and price recommendations.
- **Maintain Transaction Consistency**: Update dashboards, user balances, and property statuses automatically upon each operation.
- **Optimize Performance**: Monitor server load, scale resources, and ensure a smooth, responsive application experience.

2.2.3 Non-functional Requirements

In order to ensure the proper functioning of the decision-making system and to avoid any kind of anomaly, the implemented solution must meet a set of non-functional needs such as:

- **Maintainability**: The system must be designed for simplicity so that tasks, updates, and bug fixes can be executed with minimal complexity.
- **Evolution**: Platform administration must remain attentive to user needs and feedback, continuously enhancing the services offered while preserving the application's utility and efficiency.

- **Security**: Robust security measures are essential. The platform must enforce strong authentication protocols, access privileges, and comprehensive data encryption (both at rest and in transit). The integration of blockchain technology further ensures the immutability and integrity of sensitive information.
- **Efficiency**: The application must be effective in all circumstances, delivering prompt and reliable functionality regardless of external conditions.
- **Performance**: The system must operate optimally across diverse environments. It should consistently provide a responsive and reliable experience, even under high transaction volumes or varying network conditions.

2.3 Requirements Analysis

In this section, we'll outline the various features that our app should offer, using a general use case diagram.

2.3.1 General use case diagram

Below, we present the various actors of the application and the actions they are authorized to perform. The overall diagram is illustrated in the following figure:

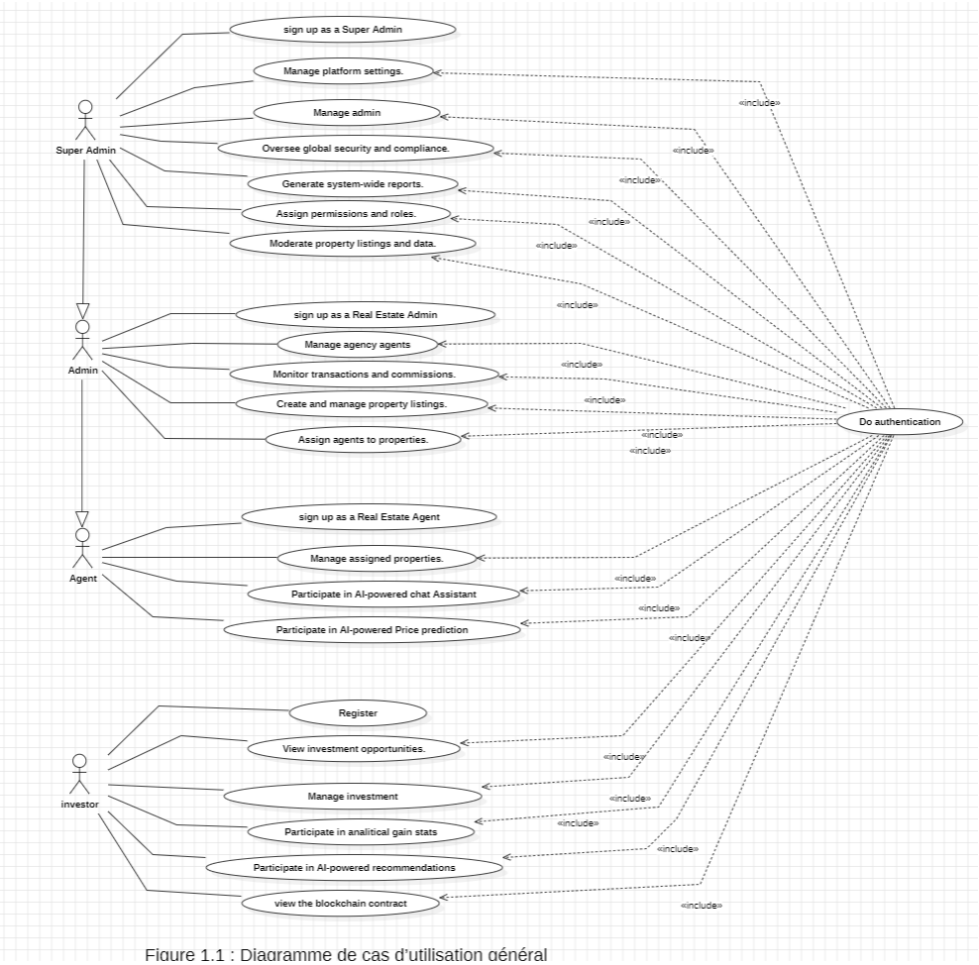

Figure 2.1: General use case diagram