DEVELOPMENT OF A WEB-BASED PROPERTY LISTING PLATFORM FOR EFFICIENT REAL ESTATE TRANSACTIONS IN NIGERIA

BY

NWAKWUE KELECHUKWU JEPHTHAH

ENG2008291

DEPARTMENT OF COMPUTER ENGINEERING

FACULTY OF ENGINEERING

UNIVERSITY OF BENIN

BENIN CITY

EDO STATE

OCTOBER, 2025

DEVELOPMENT OF A WEB-BASED PROPERTY LISTING PLATFORM FOR EFFICIENT REAL ESTATE TRANSACTIONS IN NIGERIA

BY

NWAKWUE KELECHUKWU JEPHTHAH

ENG2008291

A PROJECT SUBMITTED TO THE DEPARTMENT OF COMPUTER ENGINEERING, FACULTY OF ENGINEERING, UNIVERSITY OF BENIN

BENIN CITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF BACHELOR OF ENGINEERING (B.ENG) DEGREE IN COMPUTER ENGINEERING

OCTOBER, 2025

CERTIFICATION

This is to certify that this project, **DEVELOPMENT OF A WEB-BASED PROPERTY LISTING PLATFORM FOR EFFICIENT REAL ESTATE TRANSACTIONS IN NIGERIA** submitted by Nwakwue Kelechukwu Jephthah with matriculation number ENG2OO8291 has been examined and approved as meeting part of the requirements for the award of the degree of Bachelor of Science in Computer Engineering at University of Benin.

_____

_____

Engr. Prof. S.T Apeh Date

(Project Supervisor)

_____

_____

Engr. Dr. A.I Edeoghon Date

(Head Of Department)

DEDICATION

This project is dedicated first and foremost to Jehovah, whose unfailing love, wisdom, and guidance have sustained me from the beginning of this journey to its successful completion. Without His strength and direction, this achievement would not have been possible.

I also dedicate this work to my beloved family members, whose constant encouragement, prayers, and unwavering support have been my source of motivation throughout this period. Their belief in me gave me the courage to keep pushing forward, even when the challenges seemed overwhelming.

ACKNOWLEDGEMENT

I am deeply grateful to Jehovah God, whose boundless love, wisdom, and strength have guided me from the beginning of this journey to its successful completion. Without His divine help, none of this would have been possible.

My heartfelt appreciation goes to my beloved parents, Mr. Emmanuel Nwakwue and Mrs. Love Nwakwue, for their unwavering love, encouragement, and sacrifices throughout my academic pursuit. You have been my greatest source of motivation and strength. To my dear sister, Shulammite Nwakwue, thank you for always being a strong support and for believing in me even when times were tough. Your words of encouragement, constant check-ins, and cheerful spirit kept me going whenever I felt discouraged. You have truly been a pillar of emotional strength and inspiration.

I wish to express my sincere gratitude to my project supervisor, Prof. S. Apeh, for his continuous guidance, constructive feedback, and encouragement throughout the course of this project. His supervision, patience, and valuable insights contributed immensely to the success of this work.

I extend special thanks to my uncle, Mr. Ejayeta Tibunor, for his constant support, care, and concern. Your guidance and generosity have truly been a blessing.

I am also sincerely grateful to my friends, Efe Akhaze and Ubong Martins, for their steadfast friendship and encouragement. You were always there for me, anytime and any day, providing both moral and emotional support.

A very special appreciation goes to Oghale Igri, whose relentless effort, guidance, and encouragement during my Industrial Training and final year made a significant impact on my success. Your commitment to ensuring I gave my best is deeply appreciated.

Finally, I thank everyone who contributed in one way or another to the success of this work. Your support, directly or indirectly, has made this achievement possible.

ABSTRACT

The real estate sector in Nigeria continues to face significant challenges, including the prevalence of fraudulent property listings, unverified agents, lack of transparency, and inefficient processes for property transactions. This research proposes the development of a web-based property listing platform designed to address these issues by leveraging technology to enhance the efficiency, security, and reliability of real estate dealings. The platform aims to provide a secure environment where property seekers, verified agents, and legitimate sellers can interact, access accurate property information, and complete transactions with greater confidence.

The study follows a structured methodology, beginning with requirement gathering through stakeholder consultations, literature review, and analysis of existing platforms. The system design translates these requirements into an architecture that integrates agent verification, property document authentication, advanced search and filtering options, interactive mapping, and secure communication features. The implementation phase involves the development of a user-friendly frontend and a robust backend, followed by rigorous testing to ensure system reliability, security, and performance. Deployment and hosting will make the platform accessible to the target audience, while evaluation will measure its impact in terms of usability, trust enhancement, and fraud reduction.

The expected outcome of this research is a functional, scalable, and user-centered platform that contributes to transforming real estate transactions in Nigeria. The project is anticipated to provide a reference model for similar solutions in developing economies, demonstrating how digital technologies can promote transparency, accountability, and efficiency in informal markets. In addition to delivering a practical solution, the research aims to enrich the body of knowledge on secure web-based systems for property management, offering insights for future academic inquiry and professional practice.

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION 1

1.1 INTRODUCTION TO THE THEORY OF THE SUBJECT MATTER 1

1.2 THE PROBLEM(S) ASSOCIATED WITH THE SUBJECT OF STUDY 1

1.3 SUMMARY OF THE CURRENT STATUS OF THE PROBLEM OF THE STUDY SUBJECT 3

1.4 PROBLEM STATEMENT 4

1.5 AIM AND OBJECTIVES 5

1.6 SCOPE OF STUDY 7

1.7 RELEVANCE OF THE STUDY 8

CHAPTER 2: LITERATURE REVIEW 10

2.1 LITERATURE REVIEW 10

2.2 SUBJECT MATTER 14

2.3 RELATED WORKS 16

2.4 ENCAPSULATED RESEARCH GAPS 18

2.5 REVIEW OF THE TOOLS AND TECHNOLOGIES TO BE DEPLOYED IN THE STUDY 20

CHAPTER 3: METHODOLOGY 22

3.1 SYSTEM DESIGN APPROACH 22

3.1.1 Architectural Design 22

3.1.2 System Flow 23

3.1.3 Data Flow Diagram (DFD) 24

3.1.4 Security Considerations 25

3.1.5 System Design Goals 25

3.2 SYSTEM ARCHITECTURE 26

3.2.1 Overview of the Architecture 26

3.2.2 System Architecture Diagram 28

3.3 DATABASE DESIGN 28

3.3.6 Table Structure and Description 31

3.3.7 Relationships Between Entities 32

3.3.8 Data Integrity and Security 33

3.4 SYSTEM TOOLS AND DEVELOPMENT 33

3.4.1 Frontend Tools 33

3.4.2 Backend Tools 34

3.4.3 Development Environment 35

3.4.4 Deployment Tools 35

3.4.5 Advantages of the Chosen Tools 36

3.5 SYSTEM TESTING 37

3.5.1 Objectives of System Testing 37

3.5.2 Testing Methodology 37

3.5.3 Modules Tested 38

3.5.4 Outcome of System Testing 43

CHAPTER 4: SYSTEM IMPLEMENTATION AND RESULT 44

4.1 SYSTEM IMPLEMENTATION 44

4.2 SYSTEM REQUIREMENTS AND SETUP 45

4.2.1 Hardware Requirements 45

4.2.2 Software Requirements 46

4.2.3 System Setup Procedure 47

4.3 BACKEND IMPLEMENTATION 51

4.3.1 Backend Architecture Overview 51

4.3.2 Folder Structure of the Backend 51

4.3.3 Database Design and Table Creation 51

4.3.4 Authentication Middleware 52

4.4 FRONTEND IMPLEMENTATION 52

4.4.1 Overview of the Frontend Structure 52

4.4.2 Routing and Navigation Setup 53

4.4.3 Example of a Styled Page – Signup Page 53

4.4.4 Example of a Functional Page – My Properties Page 54

4.5 HOSTING PLAN AND DEPLOYMENT PROCESS 55

4.6 SYSTEM RESULTS AND EVALUATION 56

4.6.1 System Overview 56

4.6.2 Frontend Results 57

4.6.3 Backend Results 58

4.6.4 Database Results 58

CHAPTER 5: CONCLUSION AND RECOMMENDATIONS 59

5.1 CONCLUSION 59

5.2 CONTRIBUTION TO KNOWLEDGE 60

5.3 CHALLENGES ENCOUNTERED 60

5.4 RECOMMENDATIONS 61
5.5 FUTURE WORK 62
REFERENCES 63

TABLE OF FIGURES

Figure 2.1: An illustration of reducing transaction cost..... 14
Figure 3.1: A Use Case Data Flow Diagram..... 24
Figure 3.2: Class Diagram of the Web-Based Property Listing Platform..... 25
Figure 3.3: System Architecture..... 28
Figure 3.4: Login validation code..... 39
Figure 3.5: Backend role verification code..... 40
Figure 3.6: Code for creating property..... 41
Figure 3.7: Filter Query Code Snippet..... 42

Figure 3.8: Code To Display Uploaded Images..... 42

Figure 4.1: Developer's System Setup..... 46

Figure 4.2: Vs Code working Environment..... 47

Figure 4.3: server.js Configuration..... 48

Figure 4.4: Application Running on Browser..... 50

Figure 4.5: Backend Folder Structure..... 51

Figure 4.6: JWT Authentication Middleware..... 52

Figure 4.7: React Router Configuration for Page Navigation..... 53

Figure 4.8: Screenshot of Signup Page UI..... 54

Figure 4.9: Screenshot of My Properties Page..... 54

LIST OF TABLES

Table 3.1: Users Table..... 29

Table 3.2: Properties Table..... 29

Table 3.3: Property Images Table..... 30

Table 3.4: Transactions Table..... 30

Table 3.5: Messages Table..... 30

Table 3.6: Functional test.. 38

CHAPTER 1

BACKGROUND OF THE STUDY

1.1 INTRODUCTION TO THE THEORY OF THE SUBJECT MATTER

The real estate sector plays a critical role in the socio-economic development of every country, providing housing, business premises, and investment opportunities. In Nigeria, the real estate industry is vast but largely under-regulated in terms of digital transactions. Many property transactions are conducted through informal channels, making them prone to fraud, misrepresentation, and inefficiency.

Globally, the advent of **Property** Technology (Protech),the application of information technology to the real estate sector,has revolutionized the way property transactions are conducted. Protech solutions such as web-based property listing platforms allow for seamless property discovery, communication between stakeholders, and secure transactions. These platforms utilize modern web technologies, databases, and geolocation services to create a centralized and trusted marketplace for buyers, sellers, and agents.

Despite these technological advancements globally, Nigeria's property market has only begun to embrace digital solutions, and many existing systems fall short in ensuring trust, transparency, and user convenience.

1.2 THE PROBLEM(S) ASSOCIATED WITH THE SUBJECT OF STUDY

The real estate industry in Nigeria is vital to the economy, yet it remains largely unstructured and fraught with challenges that hinder trust, transparency, and efficiency in property transactions. The problems associated with the subject of this study are as follows:

a) **High incidence of property fraud:**

Nigeria's property market has witnessed a surge in fraudulent activities. Unsuspecting buyers or renters are often defrauded through fake property listings by non-existent agents or imposters, multiple sales or rentals of the same property to different clients, and agents misrepresenting property ownership. Many of these scams thrive because of

the absence of a reliable platform that verifies agents and sellers before they are allowed to list properties.

b) Absence of standardized agent verification:

Currently, there is no central system for authenticating real estate agents or verifying their credentials on most online platforms. Anyone can sign up and pose as an agent, making it difficult for buyers and renters to differentiate between genuine professionals and fraudsters. Lack of integration with regulatory bodies like SCUML (Special Control Unit Against Money Laundering) or REDAN (Real Estate Developers Association of Nigeria) means that platforms fail to enforce standardization and accountability among agents.

c) Inefficient property discovery process:

Finding suitable properties remains cumbersome for buyers and renters because listings are scattered across different informal sources — WhatsApp groups, social media pages, and local agents — leading to wasted time and duplicated efforts. Lack of advanced search filters or map-based search on many platforms forces users to manually sift through irrelevant listings.

d) Over-reliance on middlemen:

In the absence of a trusted digital platform, buyers and renters depend heavily on multiple intermediaries (agents, sub-agents, middlemen) who often inflate property prices and complicate negotiations. This over-reliance not only raises costs but also introduces opportunities for fraud.

e) Lack of transaction transparency:

Most existing platforms stop at just listing properties. There is no secure communication channel between buyer and agent; deals are moved offline, where fraud risk increases. There is also no provision for secure deposits, escrow services, or fraud reporting tools. No effective mechanism exists for users to flag or report suspicious listings or behavior.

f) Limited monetization models:

Current property listing platforms in Nigeria offer basic free listing services but often lack premium listing options, subscription models, or agent ranking systems that can generate sustainable revenue for the platform while offering value to agents.

1.3 SUMMARY OF THE CURRENT STATUS OF THE PROBLEM OF THE STUDY SUBJECT

Despite the growth of the real estate sector and increasing demand for property solutions in Nigeria, the problems highlighted in the subject of study remain largely unresolved. Several property listing platforms, both local and international, currently operate in Nigeria, but many of these platforms fall short in addressing the core issues of fraud prevention, agent verification, and transaction transparency.

Most existing platforms primarily function as online classified ads boards. They provide spaces for property owners, agents, or middlemen to list properties for sale or rent without enforcing strict identity checks or verifying the authenticity of listings. As a result, fraudulent listings continue to thrive, with unsuspecting buyers or renters falling victim to scams such as double sales, fake ownership claims, and inflated pricing driven by unregulated intermediaries.

Additionally, the lack of integration with regulatory bodies such as the Special Control Unit Against Money Laundering (SCUML) and the Real Estate Developers Association of Nigeria (REDAN) means that platforms have no standardized process for validating the legitimacy of agents or property developers. This gap perpetuates mistrust in online property transactions and limits the confidence of prospective buyers and renters.

Efforts by some platforms to include features like contact forms, basic user reviews, or report-listing buttons have not significantly curbed fraudulent activities. These measures are often reactive rather than proactive, as they depend on users detecting and reporting fraud after it occurs, rather than preventing fraud from happening in the first place.

Moreover, many platforms lack value-added tools such as interactive map searches, mortgage or rent calculators, or direct secure messaging features that could enhance the user experience and support informed decision-making. Monetization strategies are also weak; most platforms rely solely on basic advertising without offering premium services,

verified badges, or subscription models that can simultaneously generate revenue and improve service quality.

In summary, the current status of property listing platforms in Nigeria reveals significant gaps in trust, security, transparency, and efficiency. These gaps present an opportunity for the development of a new platform that leverages technology to solve these persistent problems in a structured and sustainable way.

1.4 PROBLEM STATEMENT

The Nigerian real estate sector continues to face critical challenges that hinder efficient, secure, and trustworthy property transactions despite the existence of various property listing platforms. A review of related works and existing platforms reveals that while several web-based solutions have been developed, they fall short in addressing the fundamental problems of agent verification, fraud prevention, and transaction transparency.

Current platforms mainly serve as online notice boards without any rigorous system to verify the legitimacy of agents, sellers, or property listings. As a result, fraudulent practices such as multiple sales of the same property, fake property ownership claims, and non-existent listings continue to plague the industry. This has led to significant financial losses and eroded public trust in online property transactions.

Several research efforts have proposed improvements, such as incorporating user reviews or report buttons, but these measures are reactive and do not adequately prevent fraud at the point of listing. The absence of integration with regulatory bodies like **SCUML** (Special Control Unit Against Money Laundering) and **REDAN** (Real Estate Developers Association of Nigeria) further limits the capacity of existing platforms to verify and standardize agent operations.

In addition, current platforms lack advanced features such as secure messaging, escrow or safe deposit services, and map-based property discovery tools that could enhance user confidence and streamline the property search process. These gaps highlight the need for a more robust, technology-driven solution that proactively ensures security, transparency, and efficiency in property transactions.

This study seeks to close these solution gaps by developing a web-based property listing platform that integrates agent verification systems linked with regulatory bodies, implements fraud prevention mechanisms at the point of listing, and provides advanced tools to facilitate safer, faster, and more reliable real estate transactions in Nigeria.

1.5 AIM AND OBJECTIVES

AIM

The aim of this study is the development of a web-based property listing platform for efficient real estate transactions in Nigeria that prioritizes security, transparency, trust, and convenience by integrating agent verification, fraud prevention measures, and advanced property discovery tools.

OBJECTIVES

a) **Design and develop a secure web-based platform for property listings:**

Build an intuitive, user-friendly interface where property owners, verified agents, and developers can register and manage property listings. Ensure that the platform architecture follows industry security standards to protect user data and prevent unauthorized access. Implement responsive design to ensure compatibility across mobile devices, tablets, and desktops.

b) **Implement a comprehensive agent verification system:**

Develop a multi-step verification process where agents must provide valid credentials, such as SCUML certificates or REDAN membership documentation. Integrate the verification process with relevant regulatory bodies (SCUML, REDAN) to cross-check and validate submitted documents. Assign verified badges to agents who pass the verification process, increasing trust among platform users. Include an admin approval stage before any agent account goes live.

c) **Provide advanced search and property discovery features:**

Design a map-based search system that enables users to find properties based on precise geographic locations. Integrate advanced filters (price range, number of

bedrooms, property type, availability, proximity to landmarks, etc.). Allow sorting of listings by newest, price (low to high, high to low), or popularity.

d) Develop fraud prevention mechanisms:

Introduce an admin-controlled listing approval system to verify listings before they are published. Provide a report/flag feature so users can report suspicious listings or agents. Monitor flagged listings and accounts through an admin dashboard, with the option to suspend or delete fraudulent content. Log suspicious activities for audit and review.

e) Create secure communication and transaction pathways:

Implement an in-platform messaging system so that buyers or renters can communicate securely with agents without needing to share personal contact details initially. Explore the feasibility of integrating escrow or secure deposit options to protect both parties during transactions.

f) Establish a sustainable monetization model:

Develop subscription plans for agents and developers that unlock premium features (e.g., more listings per month, analytics on views and leads). Provide paid options for featured or boosted listings to increase visibility. Include advertising slots for mortgage companies, property insurers, or moving services as additional revenue streams.

g) Provide decision-support tools for users:

Integrate mortgage calculators, rent calculators, and property comparison features to assist users in evaluating options. Enable users to save favorite listings and set up alerts for new listings that match their preferences.

h) Design an admin dashboard for platform management:

Build a dashboard where administrators can review and approve agent registrations, approve property listings, handle flagged reports and complaints, and view analytics on platform activity (number of listings, active users, flagged content).

i) Conduct system testing and user feedback integration:

Test the platform thoroughly for usability, security, and performance. Pilot the platform with a sample group of agents and buyers, and gather feedback for refinement before full launch.

1.6 SCOPE OF STUDY

This study focuses on the design and development of a web-based property listing platform aimed at improving real estate transactions in Nigeria by addressing issues of fraud, inefficient property discovery, and lack of trust between stakeholders. The scope covers the following key areas:

a) Agent and Developer Management

- Creation of registration and profile management systems for property agents, developers, and sellers.
- Implementation of a multi-step agent verification system (including documentation upload, admin review, and third-party validation e.g. SCUML, REDAN).

b) Property Listing and Management

- Development of interfaces that allow verified agents and property owners to create, edit, and manage property listings.
- Support for images, descriptions, pricing, location data, and key property features.
- Inclusion of admin oversight for approving listings before publication.

c) Search and Property Discovery

- Integration of advanced search filters (price range, location, property type, size, proximity to amenities, etc.).
- Map-based and list-based property discovery to enhance user experience.

d) Fraud Prevention and Reporting

- Admin tools for managing and moderating listings and agent profiles.
- User tools for reporting suspicious listings or agents.
- Mechanisms for logging and auditing flagged activities.

e) User Communication Tools

- Development of an in-platform messaging system for buyers and renters to contact agents securely without exposing personal contact information initially.

f) Monetization Features

- Design of subscription plans for agents, with tiered features (e.g., number of listings, analytics access, priority placement).
- Inclusion of paid advertising slots and featured listing options.

g) Administrative Dashboard

- A backend system for administrators to:
    - Manage users and listings.
    - Review agent verification and property approvals.
    - Monitor flagged content and system usage analytics.

h) Responsive Web Design

- Development of a platform accessible on mobile devices, tablets, and desktops for maximum reach.

i) Exclusion from scope

- This project will not cover mobile app development (native Android/iOS apps), although the web platform will be mobile-friendly.
- The platform will **not** handle direct payment processing or escrow at this stage (but will explore design readiness for future integration).

1.7 RELEVANCE OF THE STUDY

The development of a web-based property listing platform for efficient real estate transactions in Nigeria is highly relevant given the prevailing challenges in the sector. This study aims to deliver a solution that provides direct and measurable benefits to various stakeholders in the real estate industry. The relevance of the study can be highlighted as follows:

1 Enhanced transparency and trust:

The platform will help reduce fraudulent practices by introducing verified agent profiles, property approvals, and user reporting systems. This is particularly important in Nigeria's real estate market where cases of property scams and multiple sales of a single property are common. By integrating a robust verification process, the platform will build trust among buyers, sellers, renters, and agents.

2 Improved Access to Property Information

Many prospective property buyers or tenants struggle with locating available properties that meet their specific needs. The proposed platform will offer advanced search filters, map-based navigation, and detailed listings, thereby simplify property discovery and helping users make informed decisions faster.

3 Economic Impact

The platform will create new opportunities for genuine agents, developers, and property owners to showcase their offerings to a wider audience, thereby increasing sales and rental turnover. It will also generate revenue through subscription models, advertisements, and featured listings, contributing to the digital economy.

4 Support for Digital Transformation

This study promotes the adoption of digital technologies in Nigeria's real estate sector. By providing a user-friendly and mobile-responsive platform, the project aligns with national efforts to digitize business operations and enhance service delivery.

5 Beneficiaries of the Study

The study will benefit:

- **Property buyers and renters:** By providing access to verified listings and reducing the risk of fraud.
- **Agents and developers:** By offering a platform to market their properties effectively and transparently.
- **Government agencies and regulators:** Through easier monitoring of real estate transactions and trends.
- **Researchers and future developers:** By serving as a case study for building secure and scalable property tech solutions.

- **The general public:** By promoting safer real estate transactions, which can reduce disputes and economic loss

CHAPTER 2

2.1 LITERATURE REVIEW

The field of real estate has undergone significant transformation due to technological advancements, giving rise to what is known as **Property Technology (PropTech)**. This term encompasses all technological innovations aimed at streamlining property transactions, management, and investments. According to Aihie (2020), PropTech represents the necessary digital evolution for Nigeria's estate surveying and valuation professionals to remain relevant in a rapidly changing market. PropTech solutions integrate digital tools into various aspects of the real estate value chain, including property marketing, transaction management, and verification.

Web-based property listing platforms are a core component of PropTech, acting as intermediaries between property seekers (buyers or renters) and property providers (agents, developers, or individual sellers). These platforms draw upon electronic marketplace theory, where digital systems reduce search and transaction costs and provide broad access to market information (Parker & Van Alstyne, 2005). They help address the problem of information asymmetry—where buyers traditionally have less information about properties or sellers—by making detailed information publicly available (Akerlof, 1970).

The consistent growth of commerce on the internet has drawn substantial interest from stakeholders across the real estate sector. Oluwunmi and Agara (2022) note that in Lagos, technology adoption in property marketing has enhanced efficiency, allowing property seekers to access listings, virtual tours, and agent profiles at their convenience. The internet has thus become a crucial channel for connecting buyers, renters, and sellers. Obi and

Afonja (2024) argue that the integration of blockchain technology further strengthens the role of digital platforms by offering immutable records of property ownership, thereby reducing the risk of multiple ownership claims and fraud.

In the Nigerian context, platforms like **PropertyPro.ng**, **PrivateProperty.com.ng**, and **Landwey.ng** represent efforts to harness technology for property marketing and sales. However, these platforms still face significant challenges related to verification, regional data coverage, and user trust (Oluwafemi & Lawal, 2022). As highlighted by *TechCabal* (2025), while initiatives like **LandSafe** have emerged to provide property verification by connecting platforms with government records, adoption remains low, and many platforms function as basic classifieds without integrated fraud prevention or agent authentication mechanisms.

Several studies have identified how web-based property platforms contribute to addressing these challenges. Adewumi et al. (2021) proposed a multi-modal listing service that incorporates agent verification, property document checks, and integration with government registries to enhance transparency. They emphasized that these features could significantly reduce fraudulent listings and build consumer confidence. Similarly, Eze et al. (2021) highlighted that online platforms offering public visibility of agent profiles, combined with user reviews and ratings, help minimize fraudulent practices and encourage professionalism.

From a theoretical standpoint, the operation of these platforms is underpinned by **information asymmetry theory** (Akerlof, 1970), which explains the market failures that occur when one party in a transaction possesses more information than the other. In real estate, buyers often lack critical details about the property or seller, leading to poor decision-making. Web-based platforms address this by aggregating and presenting property details, ownership information, and pricing data, which helps level the playing field. **Platform theory** (Parker & Van Alstyne, 2005) further provides a foundation for

understanding how multi-sided platforms generate value by facilitating direct interaction between property providers and seekers, enhancing efficiency and trust.

The global rise in internet real estate activity is evident in statistics shared by *The Digital House Hunt: Consumer and Market Trends in Real Estate A Joint Study from The National Association of ReAlToRS® and Google*, who reported that over 90% of buyers use the internet during the home-buying process, and 75% of buyers first saw the property they eventually purchased online. Online advertising has proven to be significantly more effective than print, being up to 20 times more likely to result in a sale. These trends are mirrored in Nigeria, where although precise national statistics are lacking, urban markets show increasing reliance on online platforms for property searches (Oluwunmi & Agara, 2022).

Despite these benefits, Nigerian property seekers and providers still contend with challenges. Existing platforms often lack robust verification mechanisms, leading to cases where fraudulent agents list non-existent properties or misrepresent details (TechCabal, 2025). Moreover, regional property data is uneven, with urban centres better represented than rural areas (Oluwafemi & Lawal, 2022). The absence of a centralized property database further complicates efforts to standardize listings and verification.

Therefore, a well-designed property listing platform in Nigeria should integrate technological tools such as GIS mapping for precise location data, advanced search filters for easier property discovery, and secure communication channels that protect user data and interactions. As Adewumi et al. (2021) and Obi and Afonja (2024) suggest, coupling these features with blockchain-based verification and linkages to government property registries would significantly enhance platform credibility, promote transparency, and reduce fraud.

The theory of web-based property listing platforms combines concepts from e-commerce, information systems, and digital trust models. To describe the functioning of these platforms, various conceptual and mathematical models from literature have been proposed to explain

the interactions between stakeholders and the role of technology in addressing traditional market inefficiencies.

A typical web-based property listing platform can be described using the *multi-sided platform model*. This model demonstrates how the platform serves as an intermediary connecting two primary groups: property providers (including agents, sellers, and developers) and property seekers (buyers or renters). The platform facilitates value creation by enabling direct communication, property discovery, and transactions between these groups, as suggested in the multi-sided market theory (Parker & Van Alstyne, 2005). A descriptive illustration of this theory would show the platform at the center with arrows flowing between it and each stakeholder group, indicating the exchange of information, property data, and transactional communication.

Mathematically, platform efficiency can be expressed in terms of reducing transaction cost Tc . Following Bakos (1991), this can be modeled as:

$$Tc = Ts + Tm$$

where Ts represents search cost, and Tm represents matching cost. A web-based platform seeks to minimize both Ts and Tm by providing search filters, maps, and agent verification tools, thereby increasing market efficiency.

In terms of information asymmetry, Akerlof's (1970) model identifies the gap between what the seller knows about the property and what the buyer can observe. The platform introduces features such as verified listings, property histories, and user reviews to reduce this gap. A simplified model of information availability Ia can be written as:

$$Ia = f(Vp, Ru, Dt)$$

where Vp is verification of property, Ru is reliability of user/agent profiles, and Dt is the availability of detailed listing data. The function(f) increases as these factors improve, leading to a reduction in the risk perceived by the buyer.

Furthermore, trust can be represented using the trust-building model proposed by Gefen et al. (2003), where trust (Tr) is a function of platform security S, information quality Q, and perceived reputation P:

$$Tr = g(S, Q, P)$$

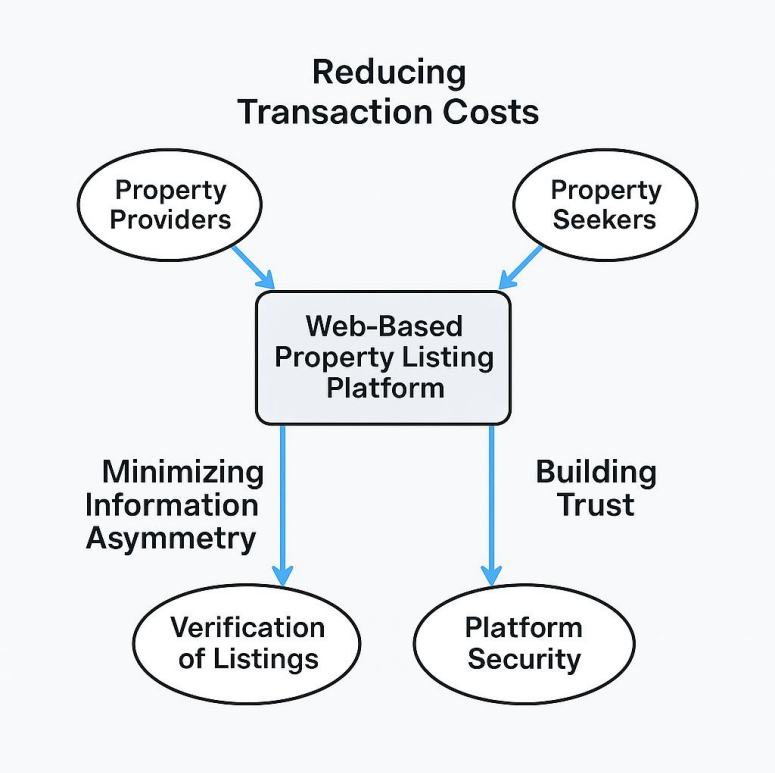


Figure 2.1 - An illustration of reducing transaction cost

A platform with strong security protocols, accurate information, and positive user feedback enhances trust, which is essential for digital transactions in real estate.

Finally, these models collectively highlight how web-based platforms can address the key challenges of traditional property markets by offering efficiency, transparency, and security, making them essential for improving real estate transactions in Nigeria.

2.2 SUBJECT MATTER

Despite the increasing adoption of web-based property listing platforms globally, their deployment and effectiveness in Nigeria face several challenges that continue to limit their full potential. These challenges arise from technological, infrastructural, regulatory, and socio-cultural factors that affect both the design and use of such platforms.

One significant challenge is the persistence of fraud and misrepresentation within the real estate sector. Many web-based platforms in Nigeria have not yet implemented robust agent and property verification systems, which leaves room for fraudulent listings and

unscrupulous agents to exploit unsuspecting buyers and renters. This undermines the primary objective of these platforms, which is to build trust and transparency. Fraudulent practices continue to affect user confidence, with many Nigerians still preferring physical transactions or relying on referrals rather than digital platforms.

Another major challenge is poor data quality and outdated property information. Many existing platforms struggle to maintain accurate and up-to-date listings due to the absence of integration with regulatory bodies and poor compliance from agents and property owners. As a result, users often encounter properties that are no longer available or have misleading descriptions, which leads to frustration and discourages repeated platform usage.

Infrastructure also poses a barrier to the widespread success of web-based property listing systems in Nigeria. Challenges such as unstable internet connectivity, low broadband penetration in rural and semi-urban areas, and limited digital literacy among some segments of the population restrict the reach and effectiveness of these platforms. Although mobile internet adoption is growing, many potential users still face difficulties accessing or navigating digital property listings effectively.

From a design perspective, existing platforms often provide limited search and discovery functionalities. Advanced search filters, geolocation services, and interactive maps are either poorly implemented or entirely absent on many Nigerian property websites. This restricts users' ability to efficiently find properties that match their preferences. Additionally, customer support and dispute resolution mechanisms are often weak or nonexistent on these platforms, leaving users without recourse in cases of dissatisfaction or fraud.

On the regulatory side, the absence of strict legal frameworks and enforcement around online property transactions has contributed to the persistence of these challenges. There is limited coordination between platforms and regulatory agencies such as the Real Estate Developers Association of Nigeria (REDAN) or SCUML, which could help enhance verification processes and ensure compliance with industry standard.

While web-based property listing platforms have the potential to revolutionize the Nigerian real estate sector, their effectiveness is currently hindered by fraud, poor data quality, infrastructure limitations, weak regulatory integration, and suboptimal user experience

features. These challenges underscore the need for innovative solutions that can address these gaps and significantly improve the trustworthiness, efficiency, and accessibility of digital property marketplaces in Nigeria.

2.3 RELATED WORKS

Several researchers and developers have explored web-based solutions aimed at improving real estate transactions in Nigeria and beyond. These related works provide useful insights into existing approaches, their strengths, and limitations, which directly inform the direction of this study.

Adewumi et al. (2021) proposed a multi-modal property listing service that integrates agent and property verification features, as well as the potential for interfacing with government registries. Their design aimed to address the longstanding issues of fraud and poor transparency that plague the Nigerian real estate market. The model also incorporated advanced search filters to assist users in refining property searches by location, price, and property features. However, the solution remained largely conceptual, with no large-scale deployment or user testing to assess real-world usability, scalability, or economic feasibility. This created a gap between theory and practice in property technology development for Nigeria.

Oluwunmi and Agara (2022) conducted an extensive study on the adoption of modern technologies in real estate marketing in Lagos. Their research highlighted the importance of tools such as GIS mapping, virtual tours, and online payment systems in enhancing property discovery and transaction efficiency. They noted that while some Lagos-based platforms had begun experimenting with these technologies, the majority relied on basic features like static listings and pin-drop location markers without interactive maps or detailed community analytics. Additionally, they found that most platforms lacked robust mechanisms for verifying property owners or agents, contributing to the persistence of fraudulent transactions. The study concluded that while awareness of PropTech solutions is

growing, the actual adoption and integration of advanced tools remain low due to infrastructural, financial, and policy barriers.

Ajayi and Omole (2017) examined urban housing and property management challenges in Nigeria, with particular focus on how technology could play a transformative role. They advocated for the integration of digital solutions to improve tenancy management, including online rent payment, contract management, and maintenance tracking. Their study emphasized that while some property managers and estate firms in Lagos and Abuja had started implementing these digital tools, many others still operated manually, leading to inefficiencies and disputes. Moreover, they argued that while these systems could help improve service delivery, their deployment had been fragmented, with no unified platform providing an end-to-end property management solution that incorporates listings, tenancy management, and ownership verification.

Oluwafemi and Lawal (2022) critically reviewed major Nigerian online property platforms such as *PropertyPro.ng*, *Nigeria Property Centre*, and *PrivateProperty.com.ng*. They observed that these platforms had succeeded in establishing themselves as popular destinations for urban property seekers, offering thousands of listings primarily focused on high-end or premium properties. However, the study identified several shortcomings. First, many of the listings on these platforms were outdated, with properties marked as available long after they had been sold or rented. Second, customer service mechanisms were often weak, with limited support for users facing disputes or fraud. Third, affordable housing and properties in peri-urban or rural areas were underrepresented. Finally, these platforms typically did not offer comprehensive verification of agents or property documents, making it easier for fraudulent actors to exploit unsuspecting users.

TechCabal (2025) reported on the emergence of *LandSafe*, a new platform aimed at tackling property verification challenges through integration with government land registries. The platform seeks to provide a reliable way for buyers and renters to confirm ownership and title status before committing to a transaction. However, the report noted that *LandSafe*

was still in its early stages, with limited integration into mainstream property portals and relatively low adoption by agents and property owners. The success of such an initiative depends heavily on government cooperation, widespread awareness, and user trust — all of which remain works in progress.

Overall, the related works show that while Nigeria has made strides in digitizing the real estate sector, significant gaps remain. These gaps include the absence of comprehensive platforms that combine listing services, tenancy management, ownership verification, advanced search functionalities, and customer support within a single system. Furthermore, many existing platforms prioritize urban and premium properties, leaving underserved markets at a disadvantage. The findings from these studies underline the need for a more holistic, inclusive, and secure property listing solution — a gap this project aims to fill.

2.4 ENCAPSULATED RESEARCH GAPS

Critically, these related works demonstrate significant progress in digitizing property transactions but also reveal recurring gaps. The most common shortcomings include the absence of robust agent and property verification systems, weak fraud detection mechanisms, limited geographic coverage beyond major cities, poor integration with regulatory frameworks, and inadequate tools for enhancing trust between users. Few systems provide comprehensive solutions that combine property listing, advanced search and filtering, geolocation, agent verification, fraud reporting, and user communication features in a single platform.

This study aims to bridge these gaps by proposing a web-based property listing platform that integrates these critical features while ensuring scalability, trustworthiness, and inclusivity for properties across various regions of Nigeria.

The review of related literature and existing systems has revealed several gaps that hinder the full realization of efficient and trustworthy web-based property listing platforms in Nigeria.

These gaps highlight critical areas where research and technological interventions are needed to improve the real estate digital ecosystem.

One key gap is the absence of a comprehensive **agent and property verification system** integrated with recognized regulatory bodies such as REDAN and SCUML. Many existing platforms fail to validate the authenticity of property owners and agents, exposing users to fraud and misrepresentation. This study intends to address this gap by proposing and implementing a multi-step verification process.

A second major gap is the **limited integration of advanced search, geolocation, and neighborhood analytics tools**. While some platforms offer basic location-based filtering, few provide interactive map features or detailed local area data (e.g., proximity to amenities, crime rates, schools). This study will address this gap by including enhanced search filters and GIS-powered property discovery.

A third gap is the **underrepresentation of affordable housing and properties outside urban centers**. Existing platforms disproportionately focus on premium properties in cities such as Lagos and Abuja, neglecting rural and peri-urban regions where housing demand also exists. This project will attempt to broaden property inclusion by encouraging listings from these underrepresented areas.

However, there are other gaps identified in the literature that fall **outside the scope of this study**:

a) Integration with national land registries or automated title verification systems. While desirable, integration with government land databases or automated title checks requires regulatory approvals and infrastructure beyond the scope of this project.
b) Development of fully mobile offline-capable apps for property listing access in areas with poor internet. This study focuses on web-based solutions optimized for low-data usage but does not address offline-first app development.
c) Incorporation of machine learning for price prediction or fraud detection. Although valuable, the deployment of AI-driven tools for these functions will be considered future work, as this study focuses on establishing a secure, functional platform baseline.

By focusing on agent verification, improved property discovery, fraud reporting, and broader property inclusion, this research will address key gaps identified in existing systems while acknowledging areas that remain open for further exploration.

2.5 REVIEW OF THE TOOLS AND TECHNOLOGIES TO BE DEPLOYED IN THE STUDY

The development of the proposed web-based property listing platform will require a suite of modern technologies, each selected for its ability to address specific aspects of the system — from user interface design to data management, security, and deployment. Below is a detailed review of the major tools and technologies intended for use:

Frontend Technologies

- HTML5 & CSS3: These will form the foundation for structuring and styling the web pages. HTML5 allows for semantic, clean markup, while CSS3 will enable the design of responsive and visually appealing layouts.
- JavaScript: This will provide interactivity and dynamic behavior to the web pages. Features such as interactive search filters, pop-ups, and form validations will rely on JavaScript.
- Frontend Framework (e.g., React.js or Vue.js): These modern JavaScript frameworks will be considered to build modular, reusable components and to ensure a seamless, fast user experience across different pages of the platform.
- Styling Framework (e.g., Bootstrap or Tailwind CSS): A CSS utility framework like Tailwind CSS or component-based library like Bootstrap will be used to accelerate the development of a clean and mobile-friendly interface.

Backend Technologies

- Server-Side Language (PHP with Laravel Framework / Python with Django Framework): These backend frameworks will handle business logic, authentication, agent verification, property management, and API integrations. Laravel (PHP) and Django (Python) are both known for their security features, scalability, and rapid development capabilities.
- Database (MySQL / PostgreSQL): A relational database management system like MySQL or PostgreSQL will be used to store structured data, including user profiles, agent verification records, property listings, transaction logs, and fraud reports.

Geolocation & Mapping

- Google Maps API / OpenStreetMap API: These will power the geolocation and interactive mapping features. They will enable users to visualize property locations on maps, filter by proximity, and assess neighborhood characteristics.

Security Technologies

- SSL/TLS for HTTPS: All communication between users and the platform will be secured using HTTPS protocols to prevent data interception.
- Password Hashing (bcrypt or Argon2): Passwords will be securely hashed before storage to protect against breaches.
- Two-Factor Authentication (2FA): Sensitive accounts such as agents and administrators will have the option of 2FA to add an extra layer of security.
- Input Validation & Web Security Best Practices: Protections will be implemented against SQL injection, cross-site scripting (XSS), and CSRF attacks to harden the platform against common vulnerabilities.

Administrative Tools

- AdminLTE / Custom React/Vue Admin Dashboard: The platform will include an admin panel to allow moderators to approve listings, handle fraud reports, and monitor platform activities through a graphical interface.

Deployment & Hosting

- Cloud Services (AWS / Microsoft Azure / DigitalOcean): Cloud hosting will be used to ensure high availability, scalability, and secure storage of platform data. These services provide tools for automatic backup, load balancing, and disaster recovery.

CHAPTER 3

METHODOLOGY

3.1 SYSTEM DESIGN APPROACH

The system design phase serves as the bridge between the project's requirements and its implementation. It focuses on transforming the conceptual framework and functional requirements into a detailed technical blueprint that defines the architecture, data flow, interfaces, and components of the web-based property listing platform. The design ensures that every module of the system works cohesively to provide secure, efficient, and user-friendly interactions across all levels like frontend, backend, and database.

Given the nature of the project, the Waterfall model was adopted as the system development methodology. This choice aligns with the structured and sequential nature of the platform's development process, where each phase (requirements gathering, design, implementation, testing, and deployment) builds upon the previous one.

3.1.1 Architectural Design

The system adopts a three-tier architecture, which separates the application into three main layers which are the presentation layer (frontend), the application layer (backend), and the data layer (database). This modular approach enhances scalability, maintainability, and security.

a) **Presentation Layer (Frontend)**

- Developed using React.js, this layer serves as the interface through which users interact with the system.
- It handles tasks such as user authentication, property search and filtering, listing visualization, and submission of data to the backend through APIs.
- The use of React components ensures code reusability, efficient state management, and smooth navigation through pages like the login/signup, property listings, property details, and user dashboard.

- Responsive design principles are implemented through **CSS3** and **Bootstrap**, ensuring the platform works seamlessly across desktops, tablets, and smartphones.

b) **Application Layer (Backend)**

- The backend, built with Node.js and Express.js, serves as the core of the system where all logic and data processing occur.
- It handles user authentication, role-based access control (admin, agent, buyer), property management (add, edit, delete), and verification processes.
- It also manages communication between the frontend and the database through RESTful APIs.
- Security measures such as JWT (JSON Web Token) authentication, input validation, and password hashing (using bcrypt) are implemented to protect user data and prevent unauthorized access.

c) **Data Layer (Database)**

- The MySQL relational database stores structured data including user details, verified agent records, property listings, and transaction logs.
- Proper table relationships (foreign keys and primary keys) are used to maintain data integrity.
- CRUD operations (Create, Read, Update, Delete) are facilitated through SQL queries and handled by the backend APIs.
- Database normalization ensures reduced redundancy and optimized data retrieval for queries such as property searches and filtering.

3.1.2 System Flow

The platform's workflow begins when a user visits the website. Depending on the role, the flow differs slightly:

- **For Buyers/Renters:**
  I. The user registers and logs in.
  II. They can browse available listings or search based on filters like price range, location, and property type.

III. They can contact agents via a secure messaging system.

- **For Agents/Developers:**
  I. Agents register and upload verification credentials (e.g., REDAN/SCUML documents).
  II. Admin reviews and approves the agent before allowing listings.
  III. Agents can then add, update, or delete property listings.
  IV. Verified agents have a “verified badge” displayed on their profiles.
- **For Administrators:**
  I. Admin logs in to the **Admin Dashboard**.
  II. Admin reviews new agent registrations and property submissions.
  III. Admin can approve, suspend, or delete accounts or listings.
  IV. Admin monitors reported listings, fraudulent activity, and overall system performance.

3.1.3 Data Flow Diagram (DFD)

To visually represent how data moves through the system, a **Data Flow Diagram (DFD)** will be included.
This diagram shows how data passes from users to the backend and database, and how responses are returned.

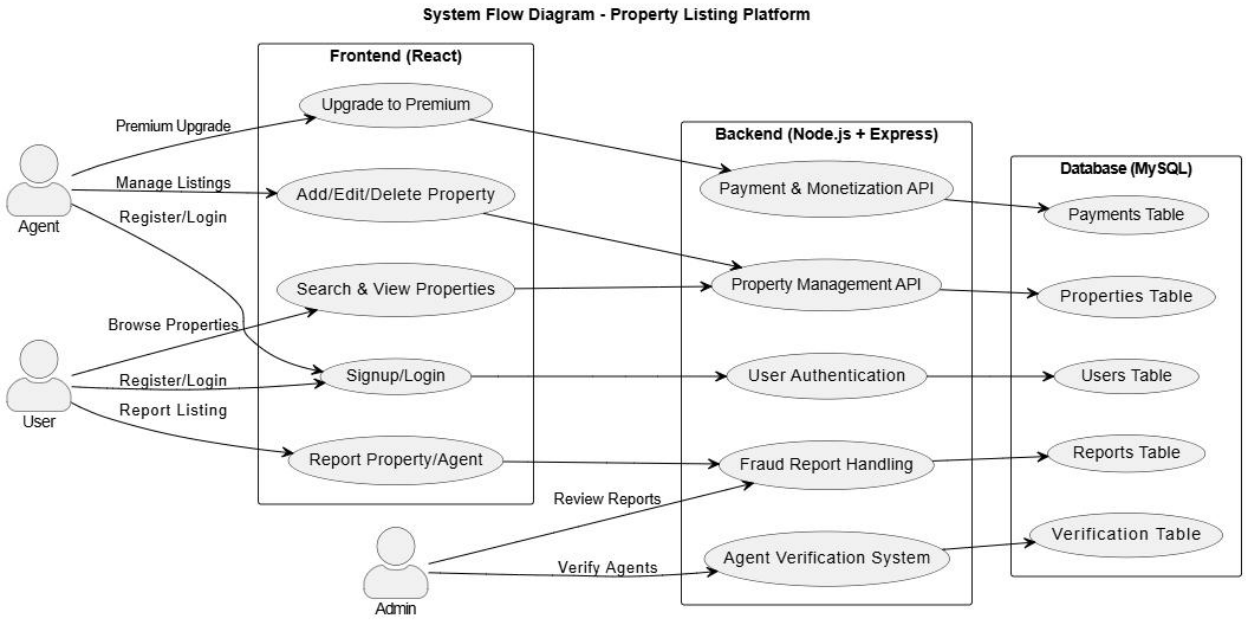


Figure 3.1- A Use Case Data Flow Diagram

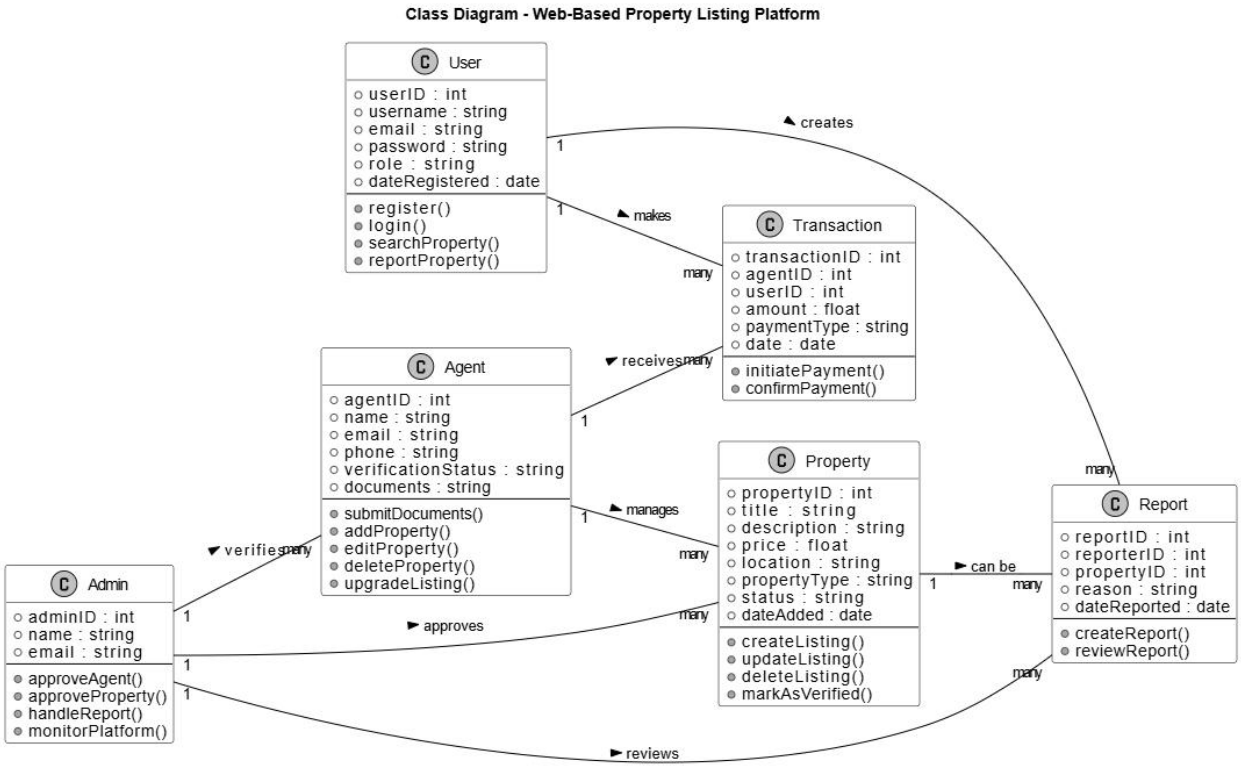


Figure 3.2 – Class Diagram of the Web-Based Property Listing Platform

3.1.4 Security Considerations

Security is a key design consideration:

- Use of HTTPS/SSL to secure communication.
- Validation and sanitization of inputs to prevent SQL Injection or XSS.
- JWT-based authentication for user session management.
- Password hashing for user data protection.
- Admin authorization for critical operations such as verification and approval.

3.1.5 System Design Goals

The design phase aims to ensure:

- Modularity : each component can be independently developed and tested.
- Scalability : future integration with APIs (e.g., Google Maps, payment gateways) is possible.
- Maintainability : clean and organized code structure enables easy updates.
- Security: all data and communication channels are protected against vulnerabilities.

3.2 SYSTEM ARCHITECTURE

The system architecture represents the **overall structure** of the *Web-Based Property Listing Platform for Efficient Real Estate Transactions in Nigeria*. It defines how various components of the system, both hardware and software, interact to achieve seamless functionality. The architecture follows a three-tier model, comprising the presentation layer (frontend), application layer (backend), and data layer (database).

This architectural approach ensures modularity, scalability, and ease of maintenance, as each layer handles specific responsibilities. The system is designed to provide an efficient interface for users (buyers, sellers, and administrators) to interact with the application, while the backend ensures proper data flow and security.

3.2.1 Overview of the Architecture

The architecture is based on the **client–server model**, where the client (user's web browser) interacts with the server (application backend), which in turn communicates with the database. Each layer is briefly described below:

a) **Presentation Layer (Frontend)**

- Implemented using React.js, this layer provides the graphical user interface (GUI) that users interact with.
- It handles user inputs such as registration, login, property search, and form submissions.
- The frontend communicates with the backend **through RESTful APIs** to send or receive data asynchronously using **Axios**.
- It ensures responsive and mobile-friendly design for accessibility on various devices.

b) **Application Layer (Backend)**

- Built with Node.js and Express.js, this layer serves as the intermediary between the frontend and the database.
- It manages all business logic, such as authentication, property verification, CRUD (Create, Read, Update, Delete) operations, and admin approvals.
- It ensures data validation, authorization, and secure communication through JSON Web Tokens (JWT).
- The backend also exposes endpoints that support user registration, login, property listing, and admin management functionalities.

c) **Data Layer (Database)**

- The MySQL relational database serves as the persistent storage layer for the platform.
- It stores user records, property details, transaction logs, and admin data.
- Data integrity is ensured using foreign keys and properly normalized tables.
- The backend interacts with the database using SQL queries through MySQL connectors integrated into Node.js.

d) **Admin Panel**

- A special section within the system that enables administrators to monitor activities, verify agents, approve listings, and manage monetization features.
- Admins access the same server but through a restricted interface protected by authentication.

e) **Hosting Environment**

- For initial deployment, the system can be hosted on GitHub Pages for the frontend and Render or Railway for the backend.

- In production, it may be deployed on AWS, Vercel, or Netlify for scalability.
- The database can be hosted on MySQL Cloud for persistent online access.

3.2.2 System Architecture Diagram

Below is a simple conceptual system architecture diagram showing the interaction among components.

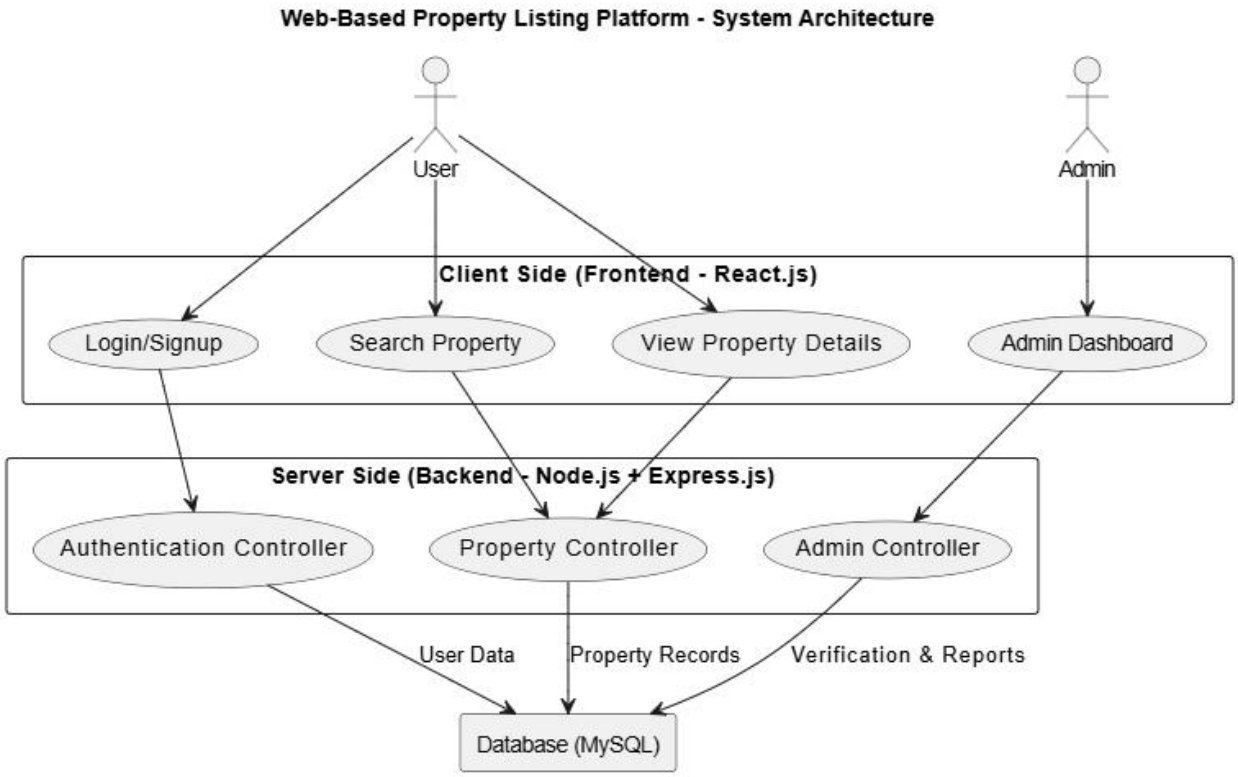


Figure 3.3 – System Architecture

3.3 DATABASE DESIGN

The database design of the Web-Based Property Listing Platform ensures efficient organization and retrieval of data related to users, properties, transactions, and messages. The design follows a relational model to maintain data integrity and eliminate redundancy. Each table is structured with well-defined relationships that support smooth operations between buyers, sellers, and administrators.

TABLE 3.1 USERS TABLE

Field	Data Type	Description
user_id	INT (Primary Key)	Unique identifier for each user
username	VARCHAR(100)	Name used by the user for login
email	VARCHAR(150)	User's email address
password	VARCHAR(255)	Encrypted user password
role	ENUM('buyer','seller','admin')	Defines the type of user
created_at	DATETIME	Timestamp when the account was created

TABLE 3.2 PROPERTIES TABLE

Field	Data Type	Description
property_id	INT (Primary Key)	Unique identifier for each property
title	VARCHAR(200)	Property title or headline
description	TEXT	Detailed description of the property
price	DECIMAL(12,2)	Cost of the property
location	VARCHAR(255)	Geographical location of the property
category	VARCHAR(100)	Type of property (e.g., flat, duplex,

		land)
status	ENUM('available','sold')	Current state of the property
seller_id	INT (Foreign Key)	References the seller from Users table

TABLE 3.3 PROPERTY IMAGES TABLE

Field	Data Type	Description
image_id	INT (Primary Key)	Unique identifier for each image
property_id	INT (Foreign Key)	Links image to property
image_url	VARCHAR(255)	Path or URL of the image file

TABLE 3.4 TRANSACTIONS TABLE

Field	Data Type	Description
transaction_id	INT (Primary Key)	Unique identifier for each transaction
buyer_id	INT (Foreign Key)	References buyer in Users table
property_id	INT (Foreign Key)	References property involved in the transaction
amount	DECIMAL(12,2)	Transaction amount
transaction_date	DATETIME	Date and time of transaction
status	ENUM('pending','completed','cancelled')	Transaction status

TABLE 3.5 MESSAGES TABLE

Field	Data Type	Description
message_id	INT (Primary Key)	Unique identifier for each message
sender_id	INT (Foreign Key)	References the sender from Users table
receiver_id	INT (Foreign Key)	References the receiver from Users table
content	TEXT	Message text content
timestamp	DATETIME	Time when the message was sent

3.3.6 Table Structure and Description

Below is a conceptual explanation of each table (you already have the tabular representation in your Word file):

1. Users Table

This table stores information about all platform users, including buyers, sellers, and agents.

- **Key fields:** user_id (PK), username, email, password, role, date_created.
- **Purpose:** Authentication, user management, and access control.

2. Properties Table

This table contains detailed information about properties listed on the platform.

- **Key fields:** property_id (PK), user_id (FK), title, description, category, location, price, status, date_added.
- **Purpose:** To store property data linked to the listing agent (user_id).

3. Property_Images Table

Stores image URLs or file paths associated with properties.

- **Key fields:** image_id (PK), property_id (FK), image_url.
- **Purpose:** To manage multiple property images without redundancy.

4. Transactions Table

Records all payments or premium upgrades made by users.

- **Key fields:** transaction_id (PK), user_id (FK), amount, transaction_type, status, date.
- **Purpose:** To track monetization and financial activities.

5. Messages Table

Handles communication between users (e.g., buyers contacting agents).

- **Key fields:** message_id (PK), sender_id (FK), receiver_id (FK), content, timestamp.
- **Purpose:** Facilitates internal messaging for negotiation or inquiries.

6. Admins Table

Stores information about the system administrators.

- **Key fields:** admin_id (PK), name, email, password.
- **Purpose:** To manage user verifications, platform moderation, and system control.

3.3.7 Relationships Between Entities

- **One-to-Many (User → Property):** Each agent can list multiple properties.
- **One-to-One (User → Agent Verification):** Each agent account has a single verification record.
- **One-to-Many (Property → Property Images):** Each property can have multiple images.
- **One-to-Many (User → Transaction):** Each user can make several payments.
- **One-to-Many (Property → Fraud Report):** Multiple reports can be filed for a single property.
- **One-to-Many (Admin → Verification/Fraud Handling):** Admins approve or resolve multiple records.

3.3.8 Data Integrity and Security

To maintain accuracy and reliability:

- **Foreign Key Constraints** ensure relational consistency.
- **ON DELETE CASCADE** is applied where necessary to maintain referential integrity.
- **Hashed Passwords (bcrypt)** are used to secure user credentials.
- **Input Validation** prevents SQL injection attacks and ensures data quality.

3.4 SYSTEM TOOLS AND DEVELOPMENT

The development of the **Web-Based Property Listing Platform for Efficient Real Estate Transactions in Nigeria** involved a careful selection of software tools, programming languages, frameworks, and development environments. These tools were chosen to ensure the system is **robust, secure, user-friendly, and scalable**, while also supporting future enhancements.

3.4.1 Frontend Tools

The frontend of the platform is the user interface, designed to allow users—buyers, sellers, and agents—to interact seamlessly with the system. The frontend tools and technologies used include:

- **React.js:** The platform was built using React.js, a widely-used JavaScript library for creating dynamic and responsive user interfaces. React's component-based architecture allows the reuse of code components, reducing development time and ensuring consistency across pages. It also enables efficient rendering, which improves user experience.
- **Pure CSS:** All styling was done using CSS3 without additional frameworks to have full control over the visual presentation. This includes responsive layouts using Flexbox and CSS Grid, as well as animations and transitions for smooth interactivity. The platform is fully mobile-friendly, supporting desktops, tablets, and smartphones.
- **Vite:** Vite was used as the frontend build tool. It provides **fast hot module replacement**, optimizing the development workflow and speeding up the build process. It is lightweight, modern, and works seamlessly with React projects.
- **Axios:** Axios, a promise-based HTTP client, was used to make API calls from the frontend to the backend. It enables asynchronous communication, ensuring smooth data exchange, such as retrieving property listings or submitting user forms.

3.4.2 Backend Tools

The backend of the system handles the business logic, authentication, data processing, and API services. The tools and technologies used include:

- **Node.js:** The platform's server-side logic was implemented in Node.js, which is event-driven and non-blocking. This makes it highly efficient for handling multiple simultaneous requests, such as user logins, property searches, and transaction processing.

- **Express.js:** Express.js was used as the backend framework to simplify routing, middleware management, and API creation. Its minimalist structure allows for scalable and maintainable code.
- **JWT (JSON Web Token):** JWT was used for secure user authentication. Upon login, the server generates a token that validates subsequent requests, ensuring that only authorized users can access certain resources or perform sensitive operations.
- **Local MySQL Database:** The platform uses MySQL as its relational database system. All user data, property listings, images, and transaction details are stored locally during development. MySQL ensures data integrity and provides efficient querying capabilities for complex operations like filtering properties by location, price, or type.

3.4.3 Development Environment

The tools and environment used during development were carefully chosen to enhance productivity, maintain code quality, and support debugging:

- **Visual Studio Code:** VS Code was the primary IDE for writing both frontend and backend code. Its extensions, such as ESLint for JavaScript and Prettier for code formatting, ensured consistent, readable, and error-free code.
- **Postman:** Postman was used to test backend API endpoints, including user authentication, property creation, and data retrieval. It allowed for validating response times, data structures, and error handling before integrating with the frontend.
- **Git and GitHub:** Git was used for version control, allowing tracking of all code changes. GitHub served as a remote repository for backup, collaboration, and later deployment.

3.4.4 Deployment Tools

The platform is planned for live deployment to ensure real-world accessibility:

- **Initial Deployment on GitHub Pages / Repository Hosting:** During the development phase, the project was hosted on GitHub to maintain version history and facilitate testing by stakeholders.
- **Future Deployment on AWS:** The system will later be deployed on AWS to provide a scalable and secure environment with high availability. AWS services will allow remote access, database management, and potential integration with cloud storage if required.

3.4.5 Advantages of the Chosen Tools

The tools and technologies selected for this project were carefully chosen to maximize efficiency, security, scalability, and maintainability. React.js was chosen for the frontend because its component-based architecture allows developers to create reusable UI components, leading to faster development and consistent design across the platform. Using pure CSS provides full control over styling, enabling the creation of responsive and visually appealing layouts without being restricted by a CSS framework. Vite was incorporated as the build tool to accelerate development through fast hot module replacement, making iterative changes immediate and seamless.

For communication between the frontend and backend, Axios was employed, offering promise-based HTTP requests that ensure smooth, asynchronous interactions with the server. Node.js serves as the backend runtime, providing a non-blocking, event-driven environment ideal for handling multiple simultaneous requests efficiently. Express.js complements Node.js by providing a lightweight framework for routing and middleware management, resulting in a clean and maintainable backend architecture. JWT was selected for user authentication, ensuring secure, token-based access that protects sensitive routes and user data.

The MySQL database was used to store all critical data, including user information, property listings, and transactions, offering strong data integrity and efficient querying capabilities. Postman was instrumental during development for testing APIs, validating data structures,

response times, and ensuring endpoints functioned correctly before frontend integration. Git and GitHub were essential for version control, enabling collaboration, tracking changes, and backing up the codebase. Finally, the planned deployment on AWS will provide a secure, scalable, and globally accessible environment, allowing the platform to accommodate multiple users and grow seamlessly over time.

3.5 SYSTEM TESTING

System testing is a critical phase of the development process in which the complete platform is evaluated to ensure that it functions correctly, securely, and efficiently under various conditions. The goal is to validate that the Web-Based Property Listing Platform for Efficient Real Estate Transactions in Nigeria meets all functional and non-functional requirements.

3.5.1 Objectives of System Testing

The objectives of system testing are:

a) To ensure that all modules of the platform work as expected individually and collectively.

b) To identify and correct errors or inconsistencies before deployment.

c) To validate data integrity, user authentication, and security measures.

d) To assess system performance, responsiveness, and usability across devices.

3.5.2 Testing Methodology

A combination of manual and automated testing strategies was employed:

- **Unit Testing:** Individual components and functions were tested separately for correctness.
- **Integration Testing:** Communication between frontend and backend modules was tested to ensure smooth data flow.
- **System Testing:** The complete platform was tested to confirm overall functionality and usability.
- **User Acceptance Testing (UAT):** Real users interacted with the platform to provide feedback on usability and experience.

The primary testing tools used were:

- **Postman:** For testing API endpoints and validating data responses.
- **Browser Developer Tools:** For debugging frontend issues, inspecting HTML/CSS, and monitoring network requests.
- **Manual Testing:** Across desktops, tablets, and mobile devices to ensure responsive design.
- **JWT Debugging Tools:** To verify token generation, expiration, and route protection.

3.5.3 Modules Tested

Below are the major modules tested with their test cases, expected results, actual results, and code snippets.

A. User Registration and Login

Test Case	Action	Expected Result	Actual Result
TC1	Register a new buyer	Successful registration, redirect to login	Passed

TC2	Register with existing email	Error message: “Email already exists”	Passed
TC3	Login with correct credentials	Access dashboard	Passed
TC4	Login with wrong password	Error message: “Invalid credentials”	Passed

TABLE 3.6 Functional test:

Code Snippet – Frontend Login Form Validation (React):

```
const handleSubmit = async (e) => {
  e.preventDefault();
  try {
    const res = await axios.post("http://localhost:5000/api/auth/login", formData);

    localStorage.setItem("token", res.data.token);
    localStorage.setItem("user", JSON.stringify(res.data.user));

    const user = res.data.user;

    if (user.role === "agent" && !user.verified) {
      navigate("/agent-profile-form");
    } else if (user.role === "agent" && user.verified) {
      navigate("/dashboard");
    } else if (user.role === "admin") {
      navigate("/admin/dashboard");
    } else {
      navigate("/");
    }
  } catch (error) {
    console.error(error);
    setMessage(error.response?.data?.message || "Login failed");
  }
};
```

Figure 3.4- login validation code

Security Testing:

- JWT token validation was tested to ensure that users cannot access secure routes without a valid token.
- Attempted direct access to /dashboard without token → correctly redirected to login.

Screenshot Reference:

- The above screenshot shows a successful login page with JWT token saved in local storage**.**

B. Role-Based Access Control (Buyer, Seller, Agent)

Functional Testing:

- Buyers can view and search properties but cannot add or delete listings.
- Sellers/Agents can add, update, or delete properties.

Code Snippet – Backend Role Verification (Node/Express):

```
101
102 const verifyRole = (role) => (req, res, next) => {
103   const userRole = req.user.role; // Extracted from JWT
104   if (userRole !== role) return res.status(403).json({ message: 'Access denied' });
105   next();
106 };
107 app.post('/api/property', verifyRole('seller'), createProperty);
108
```

Figure 3.5 – Backend role verification code

Testing Result:

- Buyer trying to access property creation page → Access denied (Passed)
- Seller successfully added property → Passed

C. Property Management (Create, Update, Delete, View)

TABLE 3.7 Functional Testing:

Test Case	Action	Expected Result	Actual Result
TC1	Add property with all required fields	Property saved in database, visible in listings	Passed
TC2	Add property with missing fields	Error message displayed	Passed
TC3	Update property details	Changes reflected in database and frontend	Passed
TC4	Delete property	Property removed from database and listings	Passed

Code Snippet – Axios Request to Create Property:

```
try {
  const data = new FormData();
  data.append("title", formData.title);
  data.append("location", formData.location);
  data.append("price", formData.price);
  data.append("description", formData.description);
  data.append("category_name", formData.category_name);

  // append multiple images
  images.forEach((image) => {
    data.append("images", image);
  });

  await axios.post("http://localhost:5000/api/properties", data, {
    headers: {
      "Content-Type": "multipart/form-data",
      Authorization: `Bearer ${token}`,
    },
  });

  setMessage("✅ Property added successfully!");
  setTimeout(() => navigate("/properties"), 1500);
} catch (error) {
  console.error(error);
  setMessage("❌ Failed to add property. Check your inputs or permissions.");
}
```

Figure 3.6- code for creating property

D. Property Search and Filter

Functional Testing:

- Users can search by location, price range, property type.
- Filters update results dynamically without page reload.

Code Snippet – Backend Property Filter Query (Node/MySQL):

```
const sql = 'SELECT * FROM properties WHERE location = ? AND price BETWEEN ? AND ?';
db.query(sql, [location, minPrice, maxPrice], (err, results) => {
  if(err) return res.status(500).json({ message: err });
  res.json(results);
});
```

Figure 3.7 – Filter Query Code Snippet

Testing Result:

- Search returned accurate results
- Filters correctly applied multiple criteria

E. Image Upload and Display

Functional Testing:

- Users can upload images when creating a property.
- Images are displayed on the property detail page.

```
const handleImageChange = (e) => {
  setImage(URL.createObjectURL(e.target.files[0]));
};
```

Figure 3.8 – Code To Display Uploaded Images

Testing Result:

- Image previews correctly
- Images stored and retrieved from local database

F. Responsiveness and Cross-Device Testing

- Tested on desktop, tablet, and mobile using browser developer tools.
- Layouts adjust correctly, images scale properly, and all buttons are accessible.
- Example: Mobile menu collapses correctly; search bar remains visible.

G. Error Handling and Security Testing

- Tested invalid login attempts, missing form fields, and invalid URLs.
- Invalid API requests return proper error messages.
- JWT expiration handled correctly → expired token redirects user to login.

3.5.4 Outcome of System Testing

The system testing phase confirmed that the platform is stable, secure, and fully functional. All modules, including user authentication, role-based access, property management, search, and image handling, performed as expected. Minor issues identified during testing were addressed, such as improving error messages and optimizing mobile layout. The system is now ready for deployment, providing a reliable, user-friendly experience.

CHAPTER 4

SYSTEM IMPLEMENTATION AND RESULT

4.1 SYSTEM IMPLEMENTATION

System implementation is the stage in which all design and planning activities are transformed into a functional and operational software system. It represents the practical realization of the design specifications developed in the previous chapters. At this stage, the Web-Based Property Listing Platform for Efficient Real Estate Transactions in Nigeria was developed using modern web technologies such as React (with Vite) for the frontend, Node.js and Express for the backend, and MySQL as the database system.

The implementation process involved setting up the development environment, coding the application modules, connecting the various components, and testing each feature to ensure it performs its intended function. Each component from the user interface to the database integration was carefully built to ensure reliability, scalability, and user-friendliness.

The project aimed to address key issues in Nigeria's real estate market such as property fraud, difficulty in verifying agents, lack of transparency in listings, and limited accessibility for property seekers. Hence, the implementation focused on providing a secure, efficient, and responsive system that allows:

- Users to register, log in, and search for properties by location.
- Agents to manage property listings, edit details, and upload property images.
- Administrators to verify registered agents, ensuring authenticity and trust.
- All users to interact with an intuitive interface on both desktop and mobile devices.

The development approach followed an incremental implementation model, where the system was built in stages. Each module such as authentication, property management, image upload, and agent verification was implemented and tested individually before being integrated into the complete platform. This approach minimized errors and made debugging more efficient.

During the course of implementation, the development tools and technologies were chosen based on their compatibility, flexibility, and efficiency. The use of React and Vite ensured fast rendering and modular component design, while Node.js provided a scalable backend for managing APIs and authentication. MySQL served as a reliable and structured database for storing user and property details.

Overall, the implementation process transformed the project from theoretical design into a functional platform that can be deployed online for real-world use. Each subsequent section of this chapter will discuss the setup, coding, integration, and testing of the system in detail, as well as present screenshots and code snippets illustrating key parts of the system's operation.

4.2 SYSTEM REQUIREMENTS AND SETUP

This section provides a comprehensive overview of both the hardware and software requirements used in developing and running the web-based property listing platform. It also explains the setup and configuration processes necessary for deploying the system on a local and live environment.

4.2.1 Hardware Requirements

The hardware requirements outline the minimum and recommended specifications for running the system efficiently. Since the project was developed using modern web technologies such as React, Node.js, and MySQL, it requires a system with adequate processing power and memory to handle development tasks and testing environments.

Component	Minimum Requirement	Recommended Requirement

Component	Minimum Requirement	Recommended Requirement
Processor	Intel Core i3 (2.4 GHz)	Intel Core i5 or higher
RAM	4 GB	8 GB or more
Storage	200 GB HDD	500 GB SSD
Operating System	Windows 10	Windows 11
Display	1280 × 720 resolution	1920 × 1080 (Full HD)

Note: The development was carried out on a Windows 11 laptop using Visual Studio Code as the integrated development environment (IDE).

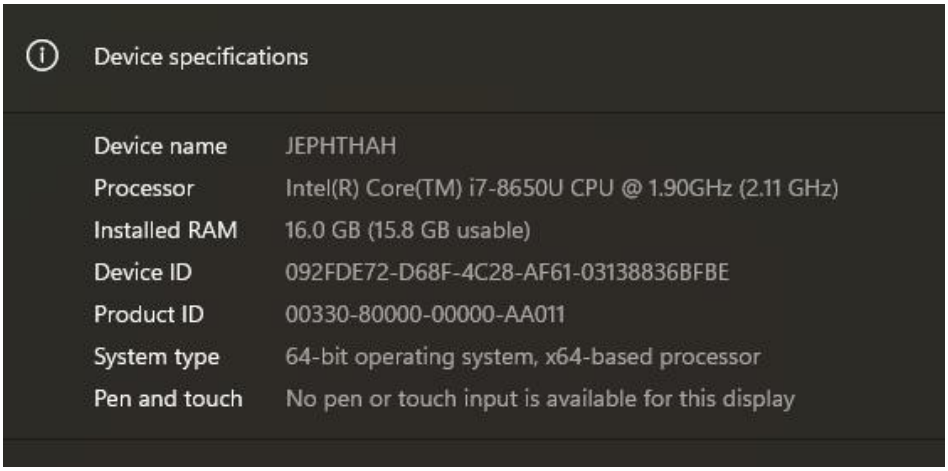


Figure 4.1 – Developer's System Setup

4.2.2 Software Requirements

The following software tools and technologies were installed and configured to support the development, testing, and hosting of the property listing platform.

Software/Tool	Purpose
Visual Studio Code (VS Code)	Main code editor used for writing frontend and backend code.

Software/Tool	Purpose
Node.js and npm	Runtime environment and package manager used for running server-side scripts and managing dependencies.
React (with Vite)	Frontend framework and build tool for developing fast, responsive user interfaces.
MySQL	Relational database management system for storing user, agent, and property data.
Postman	API testing tool for verifying backend endpoints.
JWT (JSON Web Token)	Used for secure user authentication and authorization.
Git & GitHub	Version control and cloud repository hosting platform.

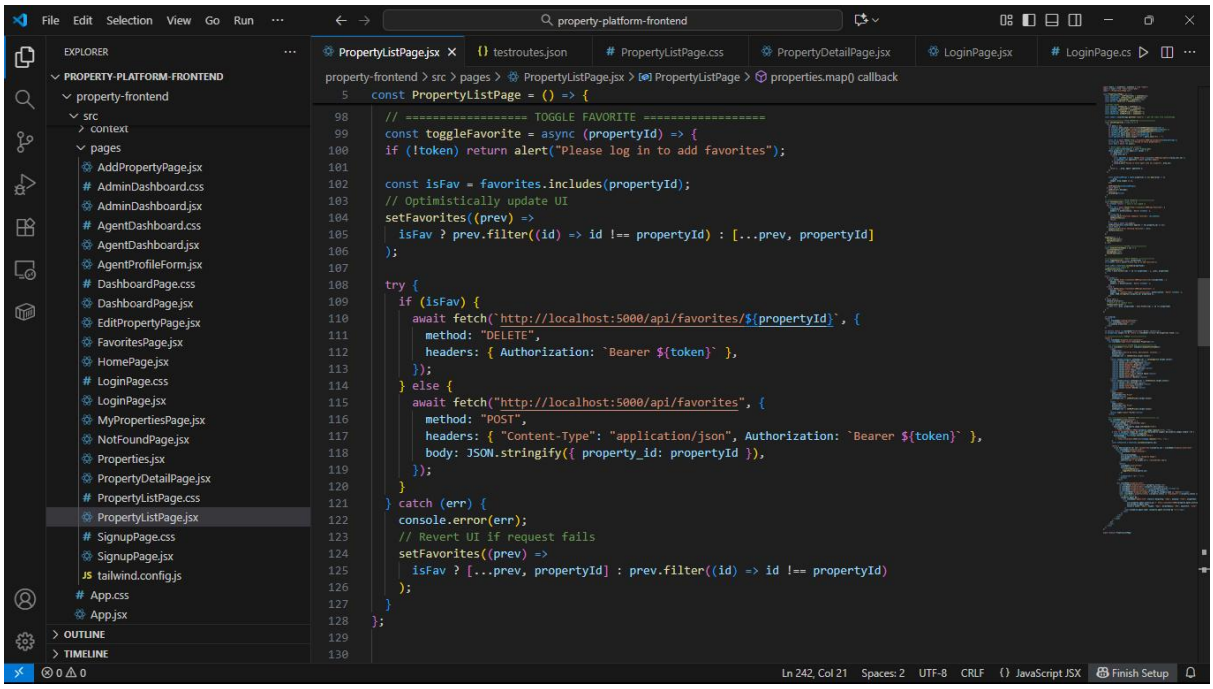


Figure 4.2 – Vs Code working Environment

4.2.3 System Setup Procedure

The following steps summarize the setup process for both the frontend and backend environments:

Step 1: Backend Setup

1. Install Node.js and MySQL on the development system.
2. Open VS Code and create a new project directory named property-platform.
3. Initialize the backend by running:

   ```
   npm init -y
   npm install express mysql2 cors dotenv bcryptjs jsonwebtoken
   ```

4. Set up the server file (e.g., server.js) and configure Express routes for:
   - User registration and login
   - Property creation, update, and deletion
   - Agent verification and filtering

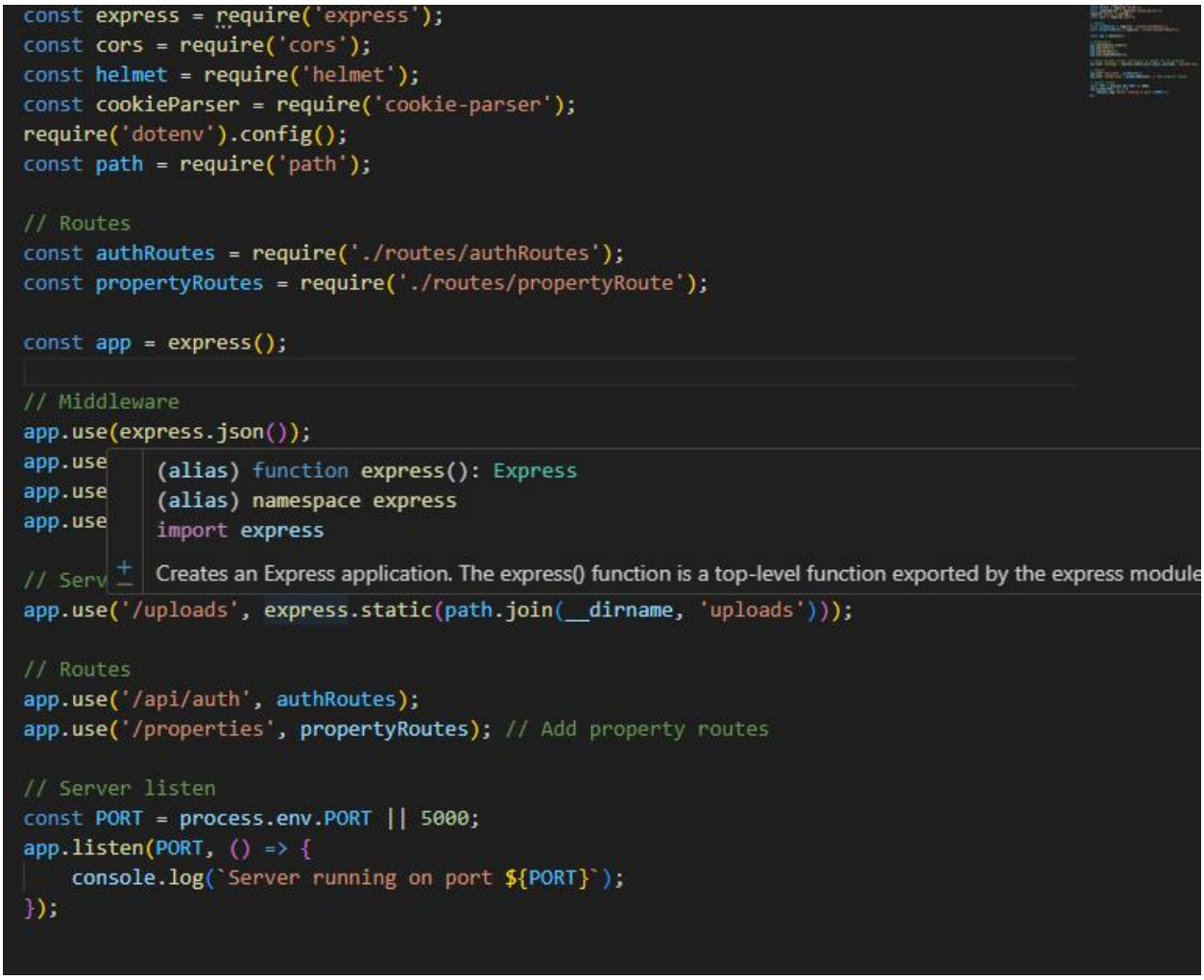


Figure 4.3 -server.js Configuration

Step 2: Frontend Setup

1. Install Vite + React for a fast frontend setup:
2. npm create vite@latest frontend --template react
3. Navigate to the frontend directory and install dependencies:

   ```
   cd property-frontend
   npm install axios react-router-dom
   ```

4. Configure vite.config.js for proxying backend requests to the Node.js server.
5. Create main components such as:
   - HomePage.jsx
   - LoginPage.jsx

- SignupPage.jsx
- MyPropertiesPage.jsx
- AdminDashboard.jsx
- AgentDashboard.jsx
- AgentProfileForm.jsx

Step 3: Database Setup

1. Launch MySQL Workbench or terminal and create the database:

```
CREATE DATABASE property_platform;
```

2. Create the required tables (see Section 4.3.2 for full SQL statements).
3. Connect the database with the backend using environment variables in a .env file:

```
DB_HOST=localhost
DB_USER=root
DB_PASSWORD=
DB_NAME=property_platform
JWT_SECRET=your_secret_key
```

Step 4: Running the Application

- Start the backend server:

```
node app.js
```

- Start the frontend:

```
npm run dev
```

- Open the browser and visit *http://localhost:5173* to view the property listing platform.

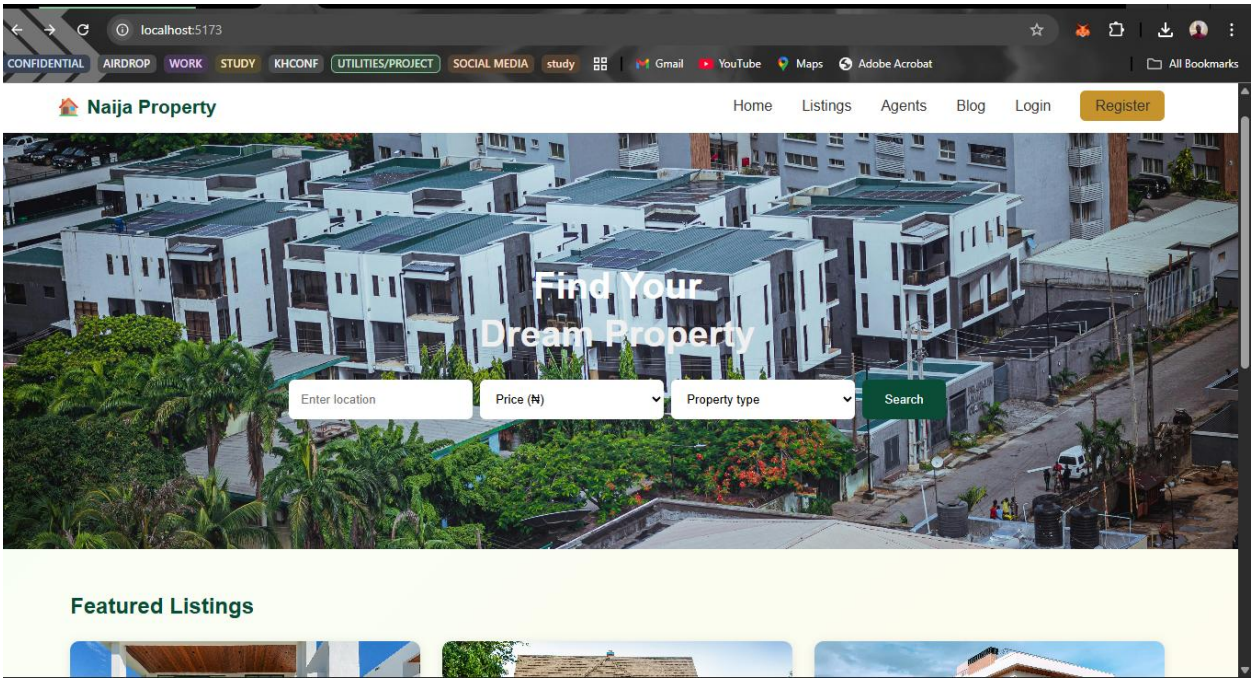


Figure 4.4 – Application Running on Browser

4.3 BACKEND IMPLEMENTATION

The backend of the property listing platform was developed using Node.js and Express.js, providing a robust and scalable server-side environment that handles all API requests, authentication, and data transactions between the frontend and the database. This section explains the backend architecture, folder structure, database design, and some representative code snippets that illustrate how major operations were implemented.

4.3.1 Backend Architecture Overview

The backend was designed using a RESTful API architecture that separates client and server responsibilities. Each route is responsible for a specific operation, such as user authentication, property management, and agent verification.

Key Backend Responsibilities:

- Handle HTTP requests from the React frontend
- Authenticate users using JWT (JSON Web Tokens)
- Perform database operations via MySQL

- Manage agent verification by the admin
- Enable property CRUD (Create, Read, Update, Delete) operations

4.3.2 Folder Structure of the Backend

The backend folder structure was organized to enhance code readability, modularity, and scalability**.**

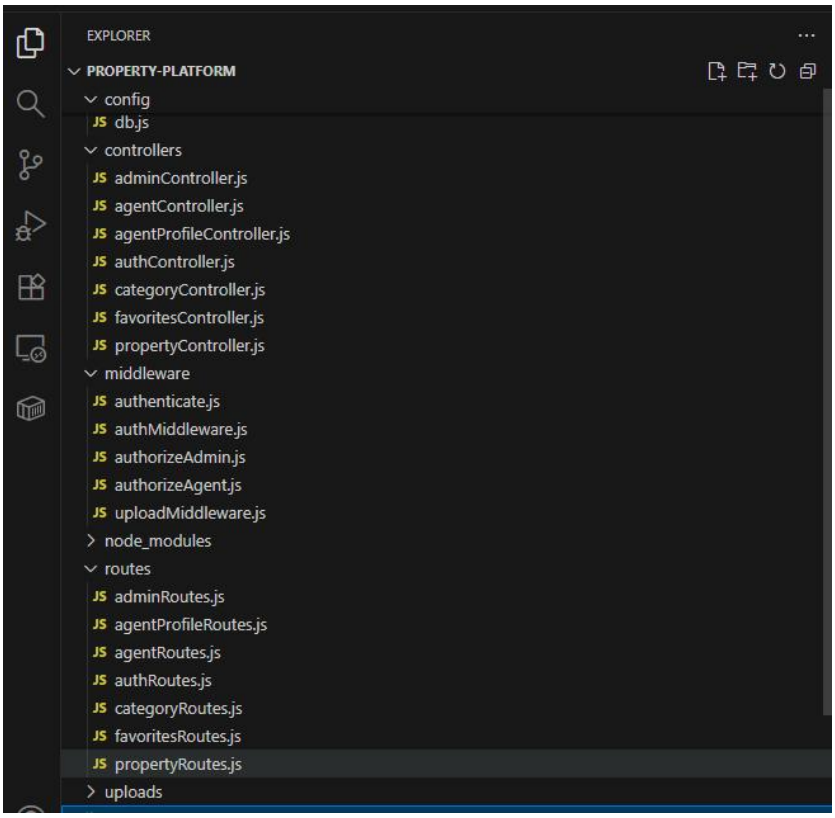


Figure 4.5 – Backend Folder Structure

4.3.3 Database Design and Table Creation

The database for the property listing platform was built using **MySQL**. It contains seven major tables:

1. users – stores details of all registered users
2. agents – stores agent information
3. agent_profiles – stores verified agent profile details
4. properties – contains property listings
5. property_images – stores property image references

6. categories – defines property types
7. favorites – stores user favorite properties

4.3.4 Authentication Middleware

To protect private routes, an authentication middleware was implemented using **JWT**. It ensures that only authenticated users (agents or admins) can perform sensitive actions.

```
const jwt = require("jsonwebtoken");

const verifyToken = (req, res, next) => {
  const token = req.headers["authorization"];
  if (!token) return res.status(403).json({ message: "Access denied" });

  try {
    const verified = jwt.verify(token.split(" ")[1], process.env.JWT_SECRET);
    req.user = verified;
    next();
  } catch (error) {
    res.status(401).json({ message: "Invalid token" });
  }
};

module.exports = verifyToken;
```

Figure 4.6 - JWT Authentication Middleware

4.4 FRONTEND IMPLEMENTATION

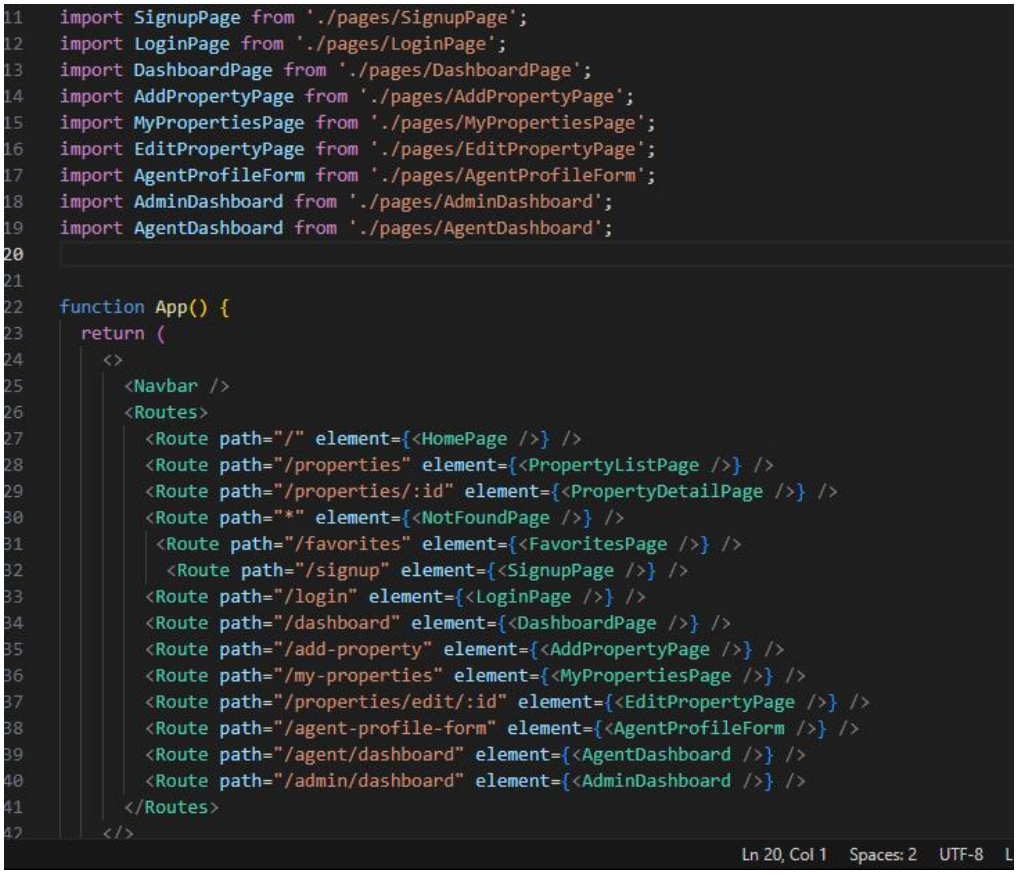
The frontend of the web-based property listing platform was developed using **React.js**, powered by **Vite** for fast development and optimized builds. The frontend serves as the interactive layer that users, agents, and administrators engage with directly. It communicates with the backend via RESTful APIs using **Axios** and provides a smooth and responsive experience through dynamic component rendering and CSS-based styling.

4.4.1 Overview of the Frontend Structure

The frontend was built with a modular component-based structure, where each page or feature is represented as an independent React component. This modularity makes the code easy to maintain and extend.

4.4.2 Routing and Navigation Setup

The navigation across pages is handled using React Router DOM, allowing seamless transitions between pages without reloading.



```jsx
import SignupPage from './pages/SignupPage';
import LoginPage from './pages/LoginPage';
import DashboardPage from './pages/DashboardPage';
import AddPropertyPage from './pages/AddPropertyPage';
import MyPropertiesPage from './pages/MyPropertiesPage';
import EditPropertyPage from './pages/EditPropertyPage';
import AgentProfileForm from './pages/AgentProfileForm';
import AdminDashboard from './pages/AdminDashboard';
import AgentDashboard from './pages/AgentDashboard';


function App() {
  return (
    <>
      <Navbar />
      <Routes>
        <Route path="/" element={<HomePage />} />
        <Route path="/properties" element={<PropertyListPage />} />
        <Route path="/properties/:id" element={<PropertyDetailPage />} />
        <Route path="*" element={<NotFoundPage />} />
        <Route path="/favorites" element={<FavoritesPage />} />
        <Route path="/signup" element={<SignupPage />} />
        <Route path="/login" element={<LoginPage />} />
        <Route path="/dashboard" element={<DashboardPage />} />
        <Route path="/add-property" element={<AddPropertyPage />} />
        <Route path="/my-properties" element={<MyPropertiesPage />} />
        <Route path="/properties/edit/:id" element={<EditPropertyPage />} />
        <Route path="/agent-profile-form" element={<AgentProfileForm />} />
        <Route path="/agent/dashboard" element={<AgentDashboard />} />
        <Route path="/admin/dashboard" element={<AdminDashboard />} />
      </Routes>
    </>
```

Ln 20, Col 1 Spaces: 2 UTF-8 L

Figure 4.7- React Router Configuration for Page Navigation

4.4.3 Example of a Styled Page – Signup Page

The Signup Page allows new users and agents to register by providing their credentials and selecting their role. Styling was achieved using pure CSS for flexibility and customization.

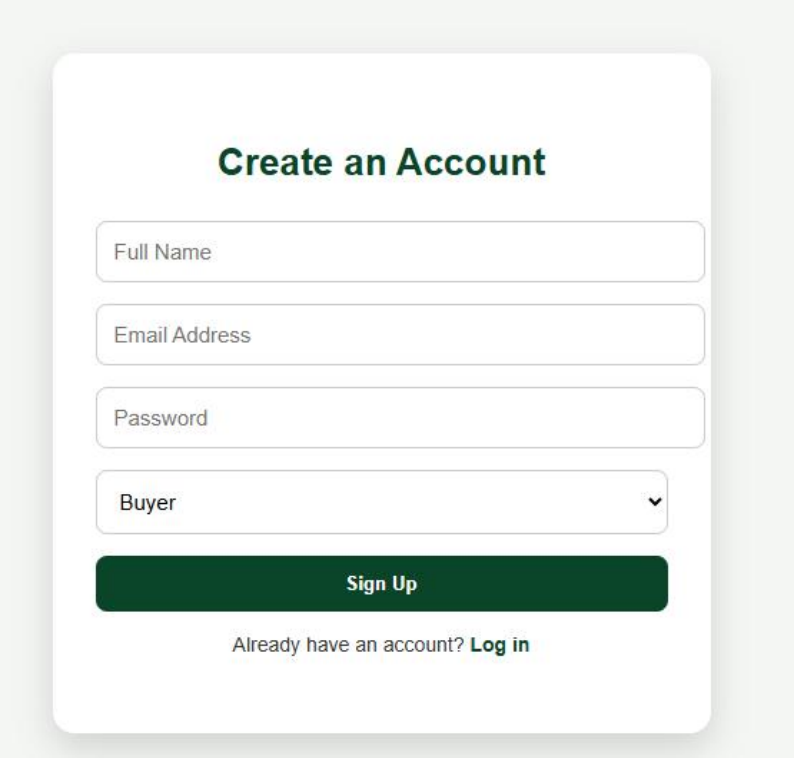


Figure 4.8 - Screenshot of Signup Page UI

4.4.4 Example of a Functional Page – My Properties Page

This page displays the list of properties posted by an agent, allowing them to edit or delete their listings. It demonstrates dynamic data fetching using Axios.

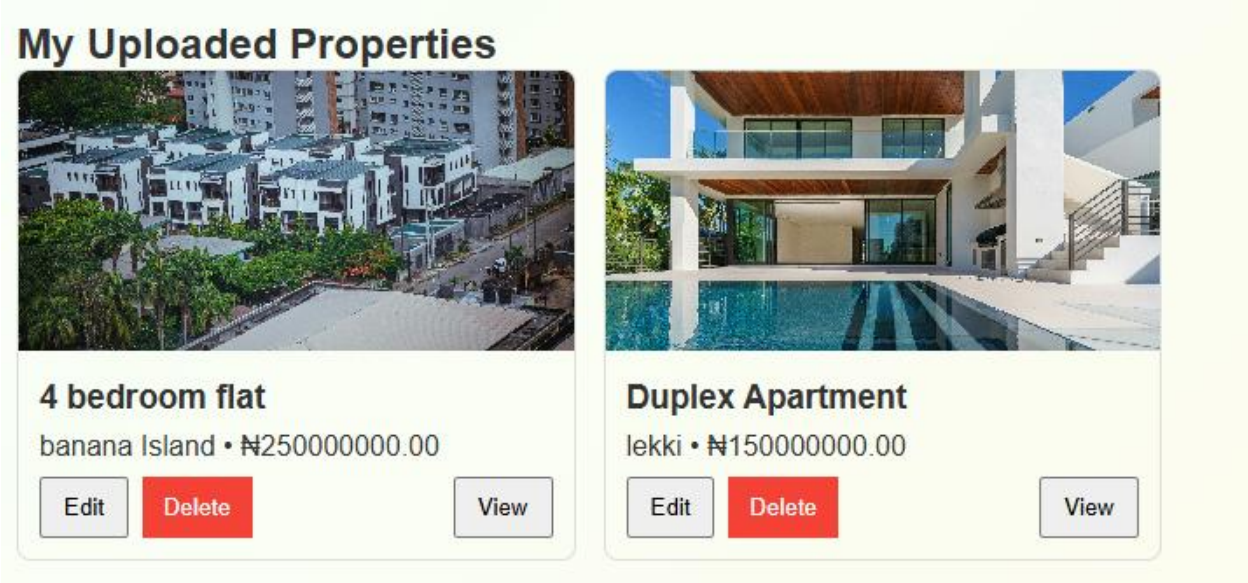


Figure 4.9 - Screenshot of My Properties Page

4.5 HOSTING PLAN AND DEPLOYMENT PROCESS

The hosting process will be executed in two stages , initial deployment on GitHub for testing and later migration to **AWS** for live production hosting.

(a) Stage 1: GitHub Hosting

GitHub is used for:

- **Version Control:** Managing source code versions and collaboration.
- **Static Hosting:** Hosting the frontend build for initial testing and accessibility.

Steps for GitHub Deployment:

1. Create a GitHub repository named property-platform.
2. Push both frontend and backend folders to separate branches (main and backend).
3. For the frontend:
4. npm run build

This command generates a production-ready dist/ folder.

5. Deploy the frontend by enabling **GitHub Pages** in repository settings.
6. Access the hosted site via https://username.github.io/property-platform.

(b) Stage 2: AWS Hosting

The long-term hosting goal is to deploy the system on Amazon Web Services (AWS) for improved performance, scalability, and reliability.

Planned AWS Services:

Service	Purpose
EC2	To host the Node.js backend server.
RDS (MySQL)	For hosting the MySQL database on the cloud.
S3	For storing property images and media files.

Planned Deployment Workflow:

1. Push code from GitHub to an AWS EC2 instance.
2. Install Node.js and MySQL on EC2.
3. Configure environment variables and run the backend using PM2 for process management.
4. Deploy the frontend (React build) to S3 with CloudFront for CDN delivery.

4.6 SYSTEM RESULTS AND EVALUATION

This section presents the results obtained after the complete implementation and integration of the Web-Based Property Listing Platform for Efficient Real Estate Transactions in Nigeria. The system was tested to ensure it met the design specifications and functional requirements outlined in earlier chapters. Each feature was carefully analyzed to verify its performance, usability, and reliability.

The system was deployed locally for testing and later prepared for live hosting through GitHub and AWS. The results from the implementation show that the system successfully achieves its objectives: enabling property seekers, agents, and administrators to interact within a secure and transparent environment.

4.6.1 System Overview

The final platform consists of three main user interfaces:

User Interface (Property Seekers):

Users can register, log in, and browse available properties. They can also search by location or category and view detailed property descriptions and images.

Agent Interface:

Agents can register, log in, and manage their listed properties. Each agent must be verified by an administrator before gaining full access to property management features.

Admin Interface:

The administrator has full control over agent verification and system management. The admin can review newly registered agents and verify their details to ensure authenticity.

Each of these interfaces is accessible through dedicated dashboard pages, which are visually distinct but consistent in layout and design.

4.6.2 Frontend Results

The frontend was designed using **React** and styled with **pure CSS** for simplicity, flexibility, and responsiveness. Pages were tested on both desktop and mobile devices to confirm adaptability.

a. Signup Page

Users and agents can create an account by providing essential information. Form validation ensures that all required fields are correctly filled before submission.

b. Login Page

Registered users log in with their email and password. Upon successful login, a **JWT token** is generated and stored in the browser's **local storage** for authentication.

c. Home Page

The homepage introduces the platform and displays featured property listings. A navigation bar allows users to access other pages easily.

d. My Properties Page

Agents can view, edit, and delete the properties they have added. The page displays properties in a clean grid format, each with a thumbnail image, title, and location.

e. Agent Dashboard

The agent dashboard provides access to functions such as adding new properties and editing their profile. It also displays verification status ("Verified" or "Pending").

f. Admin Dashboard

The admin dashboard displays newly registered agents and allows the admin to verify or reject them. Once verified, agents gain full privileges to manage their listings.

g. Agent Profile Form

Agents can update personal and professional details, such as contact information, company name, and verification documents.

4.6.3 Backend Results

The backend, implemented using Node.js and Express, handles user authentication, property management, and API communications. Each route was tested with **Postman** before being linked to the frontend.

- **User Authentication:** Secure login and registration with password hashing using bcrypt.
- **Agent Verification:** Admin routes allow updating agent verification status.
- **Property Management:** CRUD operations (Create, Read, Update, Delete) for property records.
- **Favorites Management:** Users can save and view favorite properties.

4.6.4 Database Results

The MySQL database stored user, agent, and property data efficiently. SQL tables were well normalized, ensuring data consistency and easy retrieval.

During testing, queries for adding, retrieving, and updating records executed without errors.

CHAPTER 5

5 CONCLUSION AND RECOMMENDATIONS

5.1 CONCLUSION

The successful development and implementation of the Web-Based Property Listing Platform for Efficient Real Estate Transactions in Nigeria demonstrate the practical application of modern web technologies to solve real-world problems in the real estate sector. This project was designed to address common challenges such as property fraud, agent impersonation, and lack of transaction transparency, which have affected both property buyers and sellers in Nigeria.

The system integrates three main user roles which are admin, agent, and user (property seeker) into one cohesive and secure platform. Through the use of technologies such as React, Node.js, Express, MySQL, and JWT authentication, the platform ensures a reliable and interactive experience for all users. Each component of the system from frontend design to backend processing and database management was implemented carefully to ensure functionality, security, and scalability.

The frontend, developed with React and pure CSS, provides a clean, responsive, and intuitive user interface. The backend, powered by Node.js and Express, ensures efficient handling of API routes, authentication, and communication with the MySQL database. The database itself was designed to store and manage structured data such as user details, property listings, categories, and verification statuses with integrity and consistency.

Testing and integration confirmed that all system modules work seamlessly together. Agents can register and manage their listings, administrators can verify agents to ensure authenticity, and property seekers can easily search for properties based on location and category. The system's workflow reflects a real-world property management process while maintaining high standards of data security through JWT and input validation.

Overall, the platform meets its primary objectives by:

- Providing a transparent and secure property listing environment.

- Allowing admin verification of agents to prevent fraudulent activities.
- Enabling users to easily search and explore properties.
- Maintaining a responsive design suitable for different screen sizes.

This project has proven that technology can significantly improve the reliability, accessibility, and integrity of Nigeria's real estate ecosystem when properly designed and implemented.

5.2 CONTRIBUTION TO KNOWLEDGE

This study contributes to the growing body of knowledge in web-based systems and real estate technology in the following ways:

1. It provides a model for agent verification, which can serve as a benchmark for preventing fraud in online property transactions.
2. It demonstrates how modern web technologies such as React, Node.js, and MySQL can be effectively integrated to build a scalable real estate platform.
3. It highlights the importance of secure authentication mechanisms (JWT) and data management practices in property-based applications.
4. It introduces a user-centered design approach, making real estate transactions more interactive, transparent, and efficient.

5.3 CHALLENGES ENCOUNTERED

During the development process, a few challenges were faced:

- Integration Difficulties: Connecting the frontend to the backend initially posed some issues with CORS and data handling, which were later resolved through proper middleware configuration.

- Database Optimization: Structuring tables for scalability while maintaining data integrity required careful design considerations.
- Deployment Constraints: Preparing the system for live hosting involved troubleshooting environment configurations and connection parameters.
- UI Consistency: Styling across multiple pages using pure CSS required extra effort to maintain design consistency.

Despite these challenges, the system was successfully implemented, tested, and optimized to deliver smooth functionality.

5.4 RECOMMENDATIONS

The following recommendations are made to enhance the system and guide future development:

Cloud Hosting:

The platform should be hosted on a reliable cloud service like AWS for better performance, scalability, and availability. This will also allow users to access the system remotely and store larger datasets.

Payment Integration:

Adding an online payment gateway (such as Paystack or Flutterwave) would enable users to make payments for verified property listings securely through the platform.

Chat and Notification Features:

Future versions can include real-time chat and email/SMS notifications between agents and clients to improve communication and engagement.

Advanced Search and Filters:

Implementing location-based filtering with Google Maps API or GeoJSON will make searches more accurate and user-friendly.

Security Enhancements:

Although the system already uses JWT for authentication, future improvements can include multi-factor authentication (MFA) and data encryption for higher security.

Mobile App Development:

Developing a mobile version of the platform using React Native will make the system more accessible to users who prefer mobile interactions.

User Feedback System:

Adding a feedback and rating module will help improve service quality and provide users with insights into agent credibility.

5.5 FUTURE WORK

To ensure the system's long-term sustainability, future work should focus on:

- Expanding the system architecture to support multiple administrators and agents with role-based permissions.
- Incorporating machine learning features for property recommendation and price prediction.
- Creating an analytics dashboard for tracking agent performance, property trends, and user engagement.
- Enhancing database security and introducing backup automation for data recovery.

REFERENCES

Adewumi, A. O., Iroham, C. O., Audu, D., Misra, S., & Ahuja, R. (2021). Developing a multi-modal listing service for real estate agency practice in Nigeria DOI:10.1007/978-3-030-49339-4_36. https://www.researchgate.net/publication/343482429_Developing_a_Multi-modal_Listing_Service_for_Real_Estate_Agency_Practice_in_Nigeria

Ajayi, M., & Omole, F. K. (2017). Urban housing and property management in Nigeria: Emerging challenges and solutions. *International Journal of Housing Policy, 17*(2), 217–234. https://doi.org/10.1080/14616718.2015.1130607

Akerlof, G. A. (1970). The market for "lemons": Quality uncertainty and the market mechanism. *The Quarterly Journal of Economics, 84*(3), 488–500. https://doi.org/10.2307/1879431

Bakos, Y. (1991). A strategic analysis of electronic marketplaces. *MIS Quarterly, 15*(3), 295–310. https://doi.org/10.2307/249629

Baum, A. (2017). *PropTech 3.0: The future of real estate*. Saïd Business School, University of Oxford. https://www.sbs.ox.ac.uk/sites/default/files/2018-06/PropTech3.0.pdf

Benjamin, J. D., & Chinloy, P. (1995). Performance of apartments and commercial property. *Journal of Real Estate Finance and Economics, 11*(2), 149–165. https://doi.org/10.1007/BF01096930

Bond, M. T., Seiler, M. J., & Seiler, V. L. (2000). Residential real estate brokerage web sites: An exploratory analysis of service components and site features. *Journal of Real Estate Research, 20*(3), 277–293. https://www.jstor.org/stable/24887005

Cherif, F. (2013). Real estate marketing on the internet. *Journal of Real Estate Literature, 21*(2), 261–275.

Crowston, K., & Wigand, R. T. (1999). Real estate war in cyberspace: An emerging electronic market? *International Journal of Electronic Commerce, 4*(1), 135–153. https://doi.org/10.1080/10864415.1999.11518342

James, R. N., III, McCord, M., & Montgomery, D. B. (2000). The Internet's impact on real estate brokerage. *Journal of Real Estate Portfolio Management, 6*(3), 301–311.

Oluwafemi, A., & Lawal, K. (2022). An assessment of Nigerian online property platforms: Challenges and prospects. *Journal of African Real Estate Research, 8*(2), 75–92. https://journals.uct.ac.za/index.php/JARER/article/view/1208

Oluwunmi, A. O., & Agara, E. U. (2022). Benefits and challenges to the adoption of modern technologies for real estate marketing in Lagos, Nigeria. *Journal of African Real Estate Research, 8*(1), 1–17. https://doi.org/10.15641/jarer.v8i1.1207

Parker, G., & Van Alstyne, M. (2005). Two-sided network effects: A theory of information product design. *Management Science, 51*(10), 1494–1504. https://doi.org/10.1287/mnsc.1050.0400

TechCabal. (2025, May 20). LandSafe launches to solve property verification problem in Nigeria. *TechCabal*. https://techcabal.com/2025/05/20/landsafe-property-verification-nigeria

Thrall, G. I. (1998). Adoption of GIS in real estate. *Journal of Real Estate Literature, 6*(2), 163–180.