International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 12 Issue: 04 | Apr 2025 www.irjet.net p-ISSN: 2395-0072

Enhancing House Rental Management System through User Centric Design and Technological Advancement

Ganesh Sanap[1], Dhruv pandey[2], Ayush Khopatkar[3], Sagar Gaud[4] and Dr Rohini Patil[5]

[1,2,3,4] *Student, Computer Engineering Department, Terna Engineering College, Mumbai University*
[5]*Assistant Professor, Computer Engineering Department, Terna Engineering College, Mumbai University*

Abstract - ***House rental management systems are essential in modernizing the rental process by improving efficiency and accessibility for both landlords and tenants. This research paper presents a comprehensive analysis of existing rental platforms, focusing on their core functionalities such as property listings, user authentication, communication features, and security mechanisms. Through comparative evaluation, the study identifies key limitations, including low landlord engagement, rigid user interfaces, and minimal adoption of emerging technologies. It further proposes enhancements like intuitive design elements, real-time analytics, and integration of AI and machine for dynamic rent prediction. The findings provide a foundation for developing more user-centric, adaptive, and intelligent house rental systems, aiming to enhance user satisfaction and optimize the overall rental experience.***

Key Words: *House Rental Management, Landlords, Tenants, AI and Machine Learning, Property Listing, Rental Prediction, System Performance*

1.INTRODUCTION

The house rental market is growing quickly, fueled by rising urbanization and the increasing demand from landlords and tenants for digital, streamlined solutions. House rental management systems have become essential in this space, offering platforms that handle everything from property listings and tenant screening to rent collection, maintenance tracking, and communication. These systems provide significant advantages over traditional methods, boosting efficiency, transparency, and overall satisfaction for everyone involved [1].

For tenants, these platforms make the rental process easier with features like secure online payments, automated reminders, and real-time communication. Landlords, meanwhile, gain centralized control over their properties, less administrative work, and better rent tracking and maintenance management [2]. However, even with these benefits, many current rental management systems still fall short in important areas like customization, scalability, and user experience—particularly when trying to serve a wide range of users with different levels of tech-savviness.

With the rise of modern web technologies and frameworks like the MERN stack, CodeIgniter, web scraping, and machine learning, there's now huge potential to build smarter, more user-focused systems [3]. For example, machine learning can predict rental prices and provide personalized property recommendations based on user behaviour, while web scraping can automate the updating of property listings and prices from external sources.

This research aims to thoroughly analyse existing house rental management systems, assessing their features, benefits, and limitations. By identifying gaps and exploring the potential integration of emerging technologies like AI, mobile-first design, and real-time analytics, the paper looks to uncover opportunities for future growth. The goal is to offer practical insights that can help developers, property managers, and landlords create or adopt more efficient, accessible, and smart rental management solutions that meet the needs of today's housing market.

2. LITERATURE REVIEW

A bunch of studies have looked into how house rental management systems work, what they're good at, and where they fall short. One study by Fazli et al. [1] came up with a mobile app just for student housing. The goal was to make it easier for landlords and tenants to talk, and to cut down on rental scams something students deal with a lot. They really focused on making the platform secure and pointed out that mobile apps are a solid fit for younger users who are already comfortable with tech.

Harun et al. [2] came up with a rental management system that made handling payments and keeping tenant info way more organized. Their setup showed how useful good admin tools can be for property managers stuff like tracking who's paid, sending out rent reminders, and keeping records tidy. It also helped landlords and tenants stay in the loop with each other, making the whole rental process a lot smoother.

Nair et al. [3] went a different route and focused on helping students find places to stay. They built a web app that was super easy to use, with online forms and a recommendation system to help match students with spots that actually fit what they're looking for. By using data to guide the suggestions, the app made it way quicker and easier for students to find housing that checked their boxes.

Voumick et al. [4] rolled out a web-based platform made for the rental scene in Bangladesh. They put a big focus on secure communication and making sure users could verify identities which are super important in a market where trust between landlords and tenants is a big deal. The platform also had a solid search feature, so people could easily find places that fit what they were looking for. The whole study really pushed how key security and transparency are when it comes to building trust with users.

Misyam et al. [5] came up with a rental management system aimed at making communication and transparency better between landlords and tenants. Their platform had all the essentials—property listings, user sign-ups, search tools, and even a way to handle complaints. The study pointed out how these systems aren't just for handling rent stuff, but also help with day-to-day management, making sure both sides stay on the same page and things run smoothly.

Gallera et al. [6] built an online rental system just for beach house rentals. It made managing properties and bookings way easier for owners, while also giving renters clear, up-to-date info on what was available. Since it focused specifically on vacation rentals, the study really showed how rental platforms are branching out and adapting to different markets, even the more niche ones.

Kalbande et al. [7] came up with a smart rental management system that used automation to handle stuff like payment tracking, tenant info, and generating reports. The goal was to make rental management more efficient by cutting down on manual work, giving property managers more time to focus on the bigger picture. The automation also showed how these systems could work for larger property portfolios, making them scalable as the operation grows.

Ajeokumola et al. [8] came up with a mobile, cloud-based rental management system that really focused on data security. They built in secure access controls and encryption to protect sensitive info from tenants and landlords, tackling the growing worries around digital security. The study really highlighted how important strong security features are for earning user trust and making sure rental management systems can stick around for the long haul.

Onyenwe et al. [9] built a product update-alert system that keeps track of deals and updates from e-commerce sites by pulling data like product names, prices, and posting times. While it works well, the system relies a lot on web scraping, which can be tricky if website structures change. The authors recommend adding price comparison tools to make the system even more useful.

Khatter et al. [10] came up with a web app that compares product prices across e-commerce sites using Python and Selenium. The system automates the whole comparison process, which saves users time. But, it does run into issues with error handling and depends heavily on HTML structures. To fix these problems, the authors suggest adding better error handling and using machine learning for more adaptive web scraping.

Rastogi et al. [11] built a web-based Rental House Management System to make finding a place to rent easier. Their system cut out the middleman—no more brokers—by connecting users directly with property owners, which made the whole process more transparent and efficient. It was designed to help people moving for work or school by offering an easy way to find verified rental properties. The study really showed how digital solutions can simplify house-hunting, saving time and effort when searching for the right place.

Dhamodaran et al. [12] looked at ways to improve e-rental apps by adding GPS and in-app messaging. Their research showed how these features boosted user experience and made communication between landlords and tenants easier. The study pointed out how important it is for rental platforms to be mobile-friendly and automated, showing how these tech upgrades can make things more efficient, cut down on paperwork, and simplify transactions for everyone involved.

Magno et al. [13] created an Apartment Rental Management System to handle transactions and tasks in real-time. Their system aimed to make rental property management more organized and automated, cutting down on the inefficiencies of manual record-keeping. By using a Rapid Application Development (RAD) model, they made operations smoother, billing more efficient, and communication between tenants and landlords better. The study really stressed how tech can solve common problems in rental management and highlighted the need for digital transformation in real estate.

Chiwane et al. [14] introduced a data-driven rental analysis system called EstimaRent to help with decision-making in the rental market. Their study used data analytics to give insights into trends, tenant preferences, and the best pricing strategies. They highlighted how important it is to integrate AI into rental platforms to make landlords and property managers more competitive and improve their decision-making.

Abidoye et al. [15] did a systematic review to figure out the key success factors (CSFs) for the Build-to-Rent (BTR) housing model. Their study focused on things like the role of institutional investors, affordability, and the impact of regulatory policies on making BTR projects work. They went through literature from 2011 to 2021 and came up with 32 key CSFs, with investor interest, affordability, and housing reforms being the top priorities. The research highlighted how BTR housing could help solve housing shortages and provide long-term rental options, especially for younger people dealing with affordability issues.

Srivastava et al. [16] did a deep dive into house rental websites, looking at how user-friendly they are, their security features, and how they're using AI and machine learning. Their study pointed out how online rental platforms are becoming a bigger deal for connecting landlords and tenants, while also identifying areas that need work—like improving user experience, better data encryption, and stronger fraud prevention. The research stressed the need for ongoing innovation to boost trust and security in digital rental platforms, making them more efficient and easier to use.

Rana et al. [17] built a house rental search system using Laravel, Vue.js, and PHP Machine Learning Library. Their goal was to make the rental search easier by using machine learning techniques like Linear Regression and K-Nearest Neighbors (K-NN) to give personalized property recommendations. The system also predicted rental prices and boosted search capabilities, showing just how much machine learning can bring to real estate.

Uma et al. [18] came up with an E-Housing Rental System (EHRS) to make the rental search process smoother and more automated. Their system helped house owners, people moving to new places, and agencies by offering a digital platform to list and find rentals efficiently. By using fuzzy logic and collaborative filtering, the system improved how accurate property recommendations were. The study highlighted how combining different user-focused approaches can make rental housing more accessible, all while using tech like Angular Framework and Spring Boot for better performance and scalability.

Table 1. Comparison of Existing and Proposed Features in Real Estate Platforms

Feature Category	Existing Features	Proposed Features
Listings	Residential, commercial, and land listings with details.	Enhanced Rental Listings: Include reviews and real-time availability.
Search Filters	Customizable filters and map-based search.	Advanced Filters: Add filters for lease duration, pet policies, and furnished/ unfurnished properties.
Property Comparison	Side-by-side comparison.	Advanced Comparison: Including AI for comparision
Verified Listings	Verified properties for authenticity.	Expanded Verification: Optional verification for landlords and tenants.
Notification	Price drop and new listing alerts.	Custom Price Alerts: Set specific thresholds for price drop alerts.
Tools & Calculators	EMI calculator, area converter.	Budget Tracker: Track rent and related expenses.
Communic-ation	No in app communication available	In-App Messaging: Secure communication between tenants and landlords.

The above Table 1 shows the comparison between existing system and proposed features in Real Estate Platforms [19] , [20].

3. METHADOLOGY

A. Design Methodology

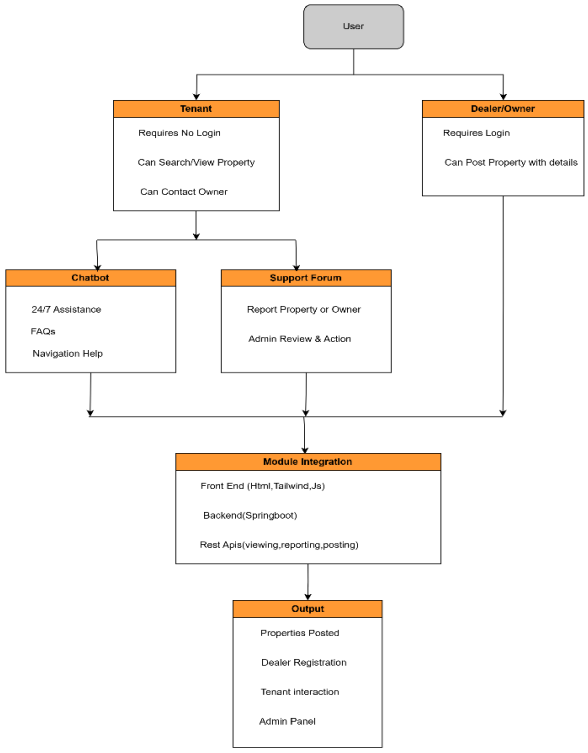

Fig.1 Proposed Methodology

Fig. 1 shows how the House Rental Management System works internally. The setup is designed to create a smooth, interactive experience for everyone like tenants, property dealers/owners, and admins by putting everything together in a modular way. The system splits the roles between tenants and dealers, so each has access to what they need. Tenants can use the platform without logging in, letting them search, view, and ask about properties directly. Dealers or property owners, on the other hand, need to log in to post their property listings, making sure everything listed is verified.

At its core, the system is built to serve two main user groups which are tenants and property owners (dealers) while keeping things easy to use, transparent, and efficient. It

starts by clearly defining these two roles. Tenants are casual users who don't need to log in. They can browse available properties, contact property owners, and even reach out with questions through built-in features. Property owners or dealers must log in before they can post listings. This login system keeps everything secure and ensures only legitimate property owners are adding properties.

To improve the experience and offer continuous help, the system has a dual-support setup with a chatbot and a support forum. The chatbot is available 24/7 to answer FAQs, guide users, and help them navigate the platform. The support forum is where users especially tenants can flag suspicious listings or dealer behavior. These reports are sent to the admin team, who take action when needed. This feedback loop makes sure the platform stays safe and responsive, giving users more trust in the system.

All of this user interaction and support runs through a strong integration module. This module is what ties everything together, combining both front-end and back-end to ensure a smooth experience. The front-end uses HTML5, Tailwind CSS, and JavaScript for a modern, responsive design that works across devices. The back-end is powered by Spring Boot, a Java-based framework that keeps everything secure, scalable, and efficient. The front-end and back-end talk to each other through RESTful APIs, handling everything from browsing properties to posting listings and filing reports. This modular setup makes the system easy to maintain and scale in the future.

At the end of the day, the system delivers several key results. Verified property listings are publicly available for tenants to browse, while dealer registration and login ensure only legitimate users can post properties. Everything tenants do from viewing listings to contacting owners is logged and managed. The admin panel allows the backend team to monitor reports, moderate content, and keep the platform in check. This setup makes the rental process smoother, ensures user satisfaction, and keeps the system scalable and reliable for the future.

B. System Analysis

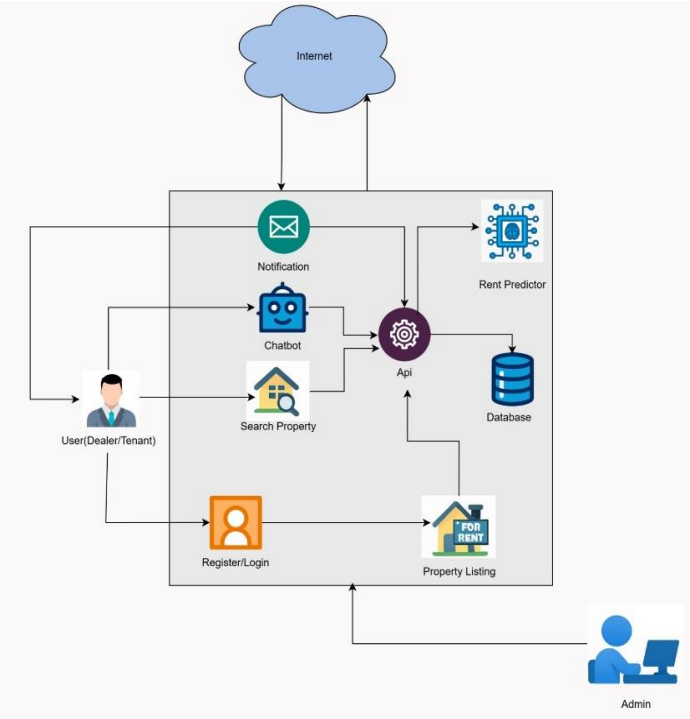

Fig.2 System Architecture

Fig.2 illustrates the system architecture for the House Rental System. The architecture is designed to support a responsive, user-friendly, and intelligent rental platform where users ,tenants, landlords (dealers), and admins can interact with the system through a series of integrated modules.

The system is accessed via the Internet, allowing seamless connectivity between users and system components.

- **User (Dealer/Tenant):**
  Users can register or log in to the system through the Register/Login module. Upon successful login, they can access other features such as property searching, listing, and chatbot support.

- **Register/Login Module:**
  This module handles authentication and user management. It allows different roles (tenant, dealer, or admin) to log in securely and access role-specific functionalities.

- **Property Listing:**
  Dealers or landlords can list rental properties through this module. These listings are stored in the Database and are made searchable for tenants.

- **Search Property:**
  Tenants can search for available properties based on filters such as location, price, and property type.

This module interacts with the API to fetch the required data from the database.

- **Chatbot Integration:**
  A chatbot is embedded within the system to offer instant assistance to users by answering queries and guiding them through available features. It communicates with the API for real-time information access.

- **Notification System:**
  The system sends notifications (e.g., property updates, verification alerts, or rental confirmations) to users through integrated email or in-app messaging services.

- **Rent Predictor:**
  This intelligent module uses machine learning or pre-trained models to analyze property data and predict suitable rental prices. It interacts with the API and database to fetch input data and return predictions.

- **API (Application Programming Interface):**
  Serving as the central communication hub, the API facilitates data exchange between the front-end modules (search, chatbot, notification) and the back-end components (database and rent predictor).

- **Database:**
  A centralized relational database stores user credentials, property listings, and reviews. It ensures data persistence and security across all system functions.

- **Admin Panel:**
  The admin has access to a backend interface to manage user activities, verify property listings, monitor system usage, and handle maintenance operations.

This modular and layered architecture ensures scalability, maintainability, and efficient data flow, while maintaining separation of concerns among different components of the system.

4. RESULT & DISCUSSION

This section presents the outcomes of the system development and evaluates its performance based on user interaction and functionality. The results demonstrate how effectively the House Rental System meets its objectives.

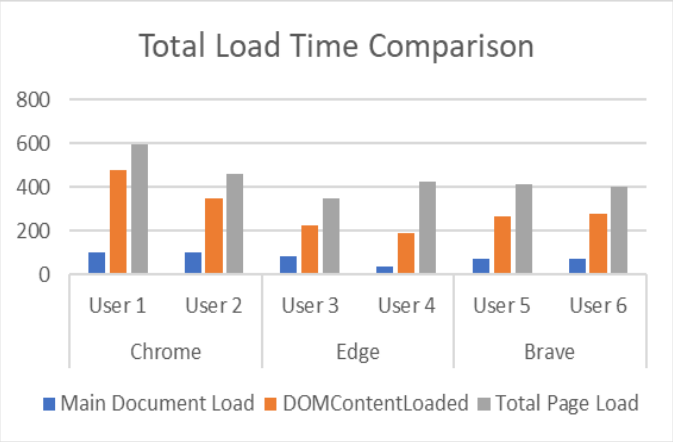

Fig.3 Total Load Time Comparison

Fig.3 shows the load time results that Chrome browsers generally have higher DOMContentLoaded and Total Page Load times compared to Edge and Brave. User 1 on Chrome experienced the highest DOMContentLoaded time, exceeding 400ms, while Edge (Users 3 and 4) consistently showed balanced and relatively lower load durations across all three metrics. Brave (Users 5 and 6) also performed efficiently, with stable Total Page Load times and moderate Main Document Load speeds, making it a competitive alternative for performance-focused users.

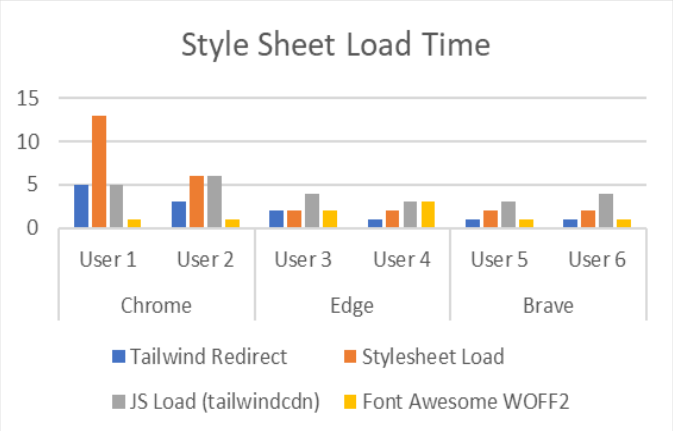

Fig.4 Style Sheet Load Time

Fig.4 shows the Style Sheet Load Time analysis highlights a notable variance, especially in the Chrome browser for User 1, where the stylesheet load time peaked around 13ms. This is significantly higher than other users across browsers. Edge and Brave showed more uniform performance, with minimal differences between Tailwind Redirect, JS Load (tailwindcdn), and Font Awesome WOFF2 times. The smoother and more consistent load behavior in Edge and Brave browsers indicates better optimization and resource handling in those environments.

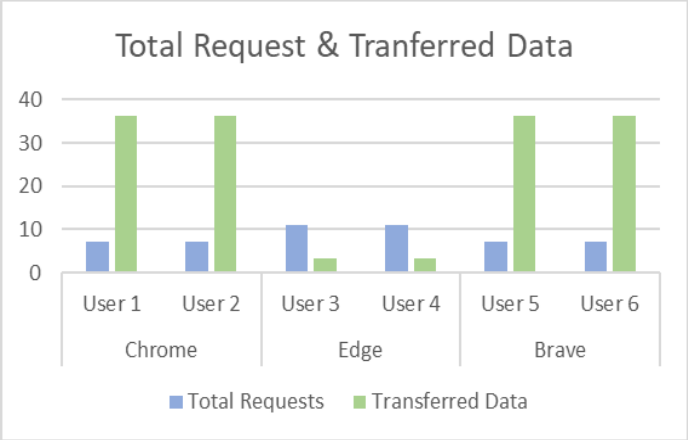

Fig.5 Total Request & Transferred Data

Fig.5 shows that ,across all users and browsers, the transferred data remained constant at around 38 units, showing minimal variation. However, the number of total requests was slightly higher in Edge (Users 3 and 4), followed by Chrome and Brave. This suggests that while data volume remained consistent, resource loading efficiency varied, with Chrome and Brave slightly outperforming Edge in minimizing the number of HTTP requests.

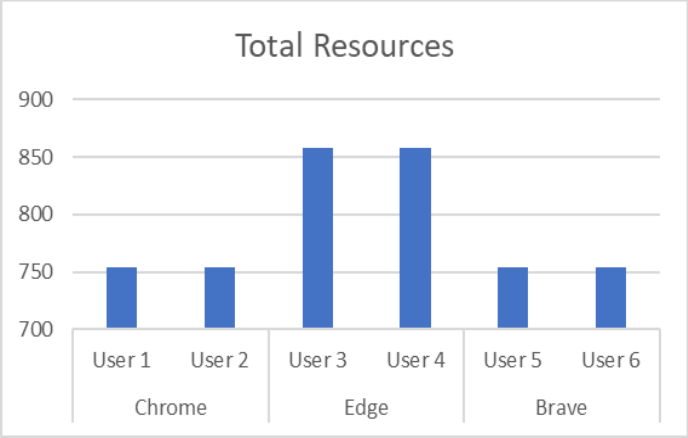

Fig.6 Total Resources

Fig.6 shows that, In terms of total resource usage, Edge browser users (User 3 and User 4) recorded the highest resource counts, exceeding 850. Chrome and Brave had lower resource usage, remaining below the 750 mark. This implies that Edge may load more supporting files or background elements, potentially impacting load time despite having efficient content loading performance.

From the analysis, it's clear that browser choice plays a big role in system performance. Chrome showed higher load times, especially with DOMContentLoaded events, likely due to its rendering or resource interpretation processes. On the other hand, Edge, despite having higher total resource counts and request rates, kept its load times low and steady, suggesting it handles resources efficiently. Brave offered a balanced performance, with quick load times, fewer requests, and lower resource usage.

Overall, these findings show that the House Rental System works reliably across modern browsers, with Edge and Brave slightly outshining in terms of resource handling and rendering efficiency. This comparison highlights the system's robustness and adaptability, making it well-suited for real-world deployment across a variety of user environments.

5. CONCLUSION

The house rental management system enhances the rental experience with features like chatbot support, real-time notifications, machine learning–based rent prediction, and a unified property listing interface. Testing across major browsers confirmed consistent performance with minimal differences in load time and resource handling.

Despite its strengths, challenges remain in boosting landlord engagement, improving UI customization, and expanding intelligent automation. Many landlords prefer traditional methods or struggle with digital tools—issues that can be eased through intuitive onboarding, simplified workflows, and clear, data-driven dashboards.

While AI features like rent prediction and chatbot support are promising, there's potential to deepen personalization with smart property recommendations, dynamic pricing, and adaptive service. Future upgrades may include blockchain for secure contracts, customizable dashboards, and PWA support for offline use—enhancing engagement, security, and accessibility for all users.

REFERENCES

[1] Muhammad Imran Fazli, & Munirah Mohd Yusof. (2023). Booking System and Procedures for UTHM House Rental Application in Parit Raja. Applied Information Technology And Computer Science, 4(2), 1471-1485.

[2] HARUN, Napisah; MOHD ZAIN, Suhaila. "Tenant Rental System Using CodeIgniter and Bulma". International Journal of Business and Technology Management, [S.l.], v. 4, n. 3, p. 340-349, oct. 2022. ISSN 2682-7646.

[3] A. Nair, A. Damle, S. Damle, and N. Kudu, "Revolutionizing Student Accommodation: A MERN-Based Rental System for Seamless Property Management," IJFMR Volume 6, Issue 3, May-June 2024. DOI 10.36948/ijfmr.2024.v06i03.19547

[4] Voumick, D., Deb, P., Sutradhar, S. and Khan, M.M. (2021) Development of Online Based Smart House Renting Web Application. Journal of Software Engineering and Applications, 14, 312-328. https://doi.org/10.4236/jsea.2021.147019

[5] Misyam, M. R., & Selamat, N. (2021). House Rental Management System. Applied Information Technology And Computer Science, 2(2), 1745-1754.

[6] Gallera, J. M. (2023). Design and evaluation of an online beach house rental system: Streamlining accommodation management and enhancing user experience. International Research Journal of Engineering and Technology (IRJET), 10(05)

[7] M. Kalbande, S. Sarkar, S. Farkase, and P. Verma, "Design and Development of Smart House Rental Management System," Journal of Property Management Technology, vol. 5, no. 2, pp. 123-135, July 2023.

[8] Ajeokumola, et al. (2020, April). A secured mobile cloud-based house rental management system. In Proceedings of the 3rd International Conference on Applied Information Technology (AIT).

[9] I. Onyenwe, E. Onyedinma, C. Nwafor, and O. Agbata, "Developing Products Update-Alert System for E-Commerce Websites Users Using HTML Data and Web Scraping Technique," International Journal on Natural Language Computing (IJNLC) Vol.10, No.5, October 2021

[10] H. Khatter, Dravid, A. Sharma and A. K. Kushwaha, "Web Scraping based Product Comparison Model for E-Commerce Websites," 2022 IEEE International Conference on Data Science and Information System (ICDSIS), Hassan, India, 2022, pp. 1-6, doi: 10.1109/ICDSIS55133.2022.9915892.

[11] Rastogi, R., et al. (2023). Rental house management system: An empirical approach with simulation. Acta Scientific Computer Sciences, 5(4), 88-102.

[12] Dhamodaran, S., Reddy, G. V. S., Saivinay, G., Gongada, P. K., Refonaa, J., & Shabu, S. L. J. (n.d.). House rental application system. In Article 53. https://doi.org/10.13052/rp-9788770040723.053

[13] Magno, J. M., Pelayo, R. S., & Soberano, K. T. (2024). House rental system research. International Journal of Multidisciplinary Research and Analysis, 7(7). https://doi.org/10.47191/ijmra/v7-i07-29

[14] Chiwhane, S., Bagane, P., Sourabh, A., Nagrale, S., Jha, S., & Pandey, S. (2023). EstimaRent: Data driven rental housing optimization and market analysis for enhanced decision-making. International Journal of Intelligent Systems and Applications in Engineering, 12(2), 20-28. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/4216

[15] Abidoye, R., Ayub, B., & Ullah, F. (2022). Systematic literature review to identify the critical success factors of the build-to-rent housing model. Buildings, 12(171). https://doi.org/10.3390/buildings12020171

[16] RENT X: House and home rental websites – A comprehensive analysis. (2023). International Journal of Innovative Research in Technology, 11(6), 2274-2279. Retrieved from https://ijirt.org/Article?manuscript=169792

[17] Rana, M. S., Omar, S. I., & Pranta, M. F. M. (2024). House rental finding system using Laravel, Vue.js, and PHP machine learning library. IOSR Journal of Computer Engineering, 26(5), 8-12. https://doi.org/10.9790/0661-2605010812

[18] Uma, P., Rao, M. S., Balaji, M. J., Kalyan, R. P., & Mahesh, P. V. (2022). E-housing rental system using hybrid user-centric private or public property recommender system using fuzzy logic and item-based collaborative filtering. International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 8(3), 354-358. https://doi.org/10.32628/CSEIT228383

[19] 99acres.com. (n.d.). Property in India, Real Estate Property for Sale/Rent in India. Retrieved September 20, 2024, from https://www.99acres.com/

[20] MagicBricks.com. (n.d.). Real Estate | Property in India | Buy/Sell/Rent Properties. Retrieved October 17, 2024, from https://www.magicbricks.com/

[21] Yuan, J., Li, W., Xia, B., Chen, Y., & Skibniewski, M. J. (2019, June 26). OPERATION PERFORMANCE MEASUREMENT OF PUBLIC RENTAL HOUSING DELIVERY BY PPPS sss WITH FUZZY-AHP COMPREHENSIVE EVALUATION. International Journal of Strategic Property Management. https://doi.org/10.3846/ijspm.2019.9820

[22] Kim, S. K., & Huh, J. (2020, October 27). Autochain platform: expert automatic algorithm Blockchain technology for house rental dApp image application model. Eurasip Journal on Image and Video Processing. https://doi.org/10.1186/s13640 020-00537-z

[23] Gill, R., Kalwar, S., & Memon, I. A. (2021, January 5). Yeh's Satisfaction Index Modelling of Tenants in Rental Appartments (A Case Study of Latifabad Hyderabad). Sukkur IBA Journal of Computing and Mathematical Sciences. https://doi.org/10.30537/sjcms.v4i2.654

[24] Sidhu, N., Pons-Buttazzo, A., Muñoz, A., & Terroso-Sáenz, F. (2021, July 1). A Collaborative Application for Assisting the Management of Household Plastic Waste through Smart Bins: A Case of Study in the Philippines. Sensors. https://doi.org/10.3390/s21134534

[25] S. Satapathy, S. M., Jhaveria, R., Khanna, U., & Dwivedi, A. K. (2014). Smart Rent Portal Using Recommendation System Visualized By Augmented Reality. "Procedia

Computer Science", 197-206. Retrieved From Www.Elsevier.Com/Locate/Procedia

[26] Afzal, S., Rouf, T., Qadir, S., & Shah, S. (2021). Online Rental Housing. "Journal Of Emerging Technologies And Innovative Research (Jetir)", 8(11). Retrieved From Https://Www.Jetir.Org

[27] Misyam, M.R., et al. (2021). House Rental Application System. River Publishers. An article which considers web application of houses being implemented in the renting procedure by people of Bangladesh to tenants and landlords

[28]Rathore K, et al. The Rental Zone (House Renting Website). International Research Journal of Modernization in Engineering Technology and Science, 4(8), 2243-2251. It brings a paper on the importance of the online house renting system against the backdrop of the demand for houses in cities.

[29] Zeng Y., & Chen H., (2019). Methods of evaluating user experiences for online rental platforms: International Information Systems Review.

[30]Setyarsih, E. (2017). Online Rental Housing Management System. Journal Article on Housing Management Systems that discusses various existing systems for house renting web applications.