www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 01, January 2025, pp : 1771-1775

e-ISSN : 2583-1062
Impact Factor : 7.001

CLEAN IT (YOUR TASK YOUR GIG)

Prof. Poonam Siddarkar[*1], Ahmed Pasha[*2], Unsiya Zafreen[*3], Rajvi Chakraborty[*4], MD Faizuddin[*5]

[*1]Associate Professor, Information Science And Engineering, AMC Engineering College, Bengaluru, Karnataka, India.

[*2,3,4,5]Students, Information Science And Engineering, AMC Engineering College, Bengaluru, Karnataka, India.

ABSTRACT

This research investigates a gig-based platform that connects individuals with freelance workers for various on-demand tasks, including purchasing goods and waste collection. The platform employs modern technologies, such as a React.js and Tailwind CSS-based mobile application, with JWT verification to enhance accuracy and response times. Secure transactions are facilitated through a payment gateway. To promote efficiency and community engagement, challenges, and recycler partnerships are integrated. The study highlights how the gig economy can improve service delivery while offering flexible earning opportunities. Through a seamless backend and user-friendly interface, the platform streamlines task management, fostering a cleaner and more organized environment.

Keywords: Gig Economy, Task Management, JWT Authentication, Mobile Application, Payment Gateway, Sustainability.

1. INTRODUCTION

In today's fast-paced world, accessing efficient and reliable services for everyday tasks, such as purchasing items or waste management, can be challenging. Traditional systems often fail to meet the increasing demand for flexible and on-demand solutions. The rise of the gig economy has bridged this gap by connecting users with freelance workers who can provide various services efficiently.

This research introduces a gig-based platform designed to connect users with workers for tasks ranging from waste collection to purchasing goods. By leveraging mobile applications, and secure payment systems, the platform ensures accuracy, reliability, and user convenience. Additionally, features like leaderboards encourage community engagement, while partnerships with recyclers promote sustainability. This integrated approach creates an eco-friendly and user-centric service model, demonstrating the gig economy's potential to enhance service accessibility and efficiency.

2. LITERATURE SURVEY

Brynjolfsson and McAfee (2014) examined the impact of advanced technologies on the gig economy in The Second Machine Age. They discussed how the rise of automation and AI is transforming industries and labor markets. However, their study does not focus on gig economy platforms or their operational systems, offering broader insights into technological shifts. Lindhjem et al. (2019) explored the role of technology in improving waste management systems. Their research focused on the application of smart solutions for waste collection, aligning with Clean-It's goals of improving task efficiency. However, it did not address gig-based platforms or freelance task management. Sharma et al. (2020) explored leveraging AI for waste management, focusing on its potential for optimizing resource usage. Their study highlighted AI's role in environmental sustainability but did not apply to Clean-It, which uses simpler, non-AI technologies. Sun et al. (2021) researched machine learning for task verification in gig platforms, emphasizing its importance in ensuring task accuracy. This is relevant to Clean-It, but it uses JWT tokens for user authentication instead of machine learning. Xu et al. (2022) proposed AI-based optimization in gig economy platforms, but their research on AI-driven service delivery is not applicable to Clean-It, which relies on simpler technologies for scalability.

3. METHODOLOGY

The methodology for the clean-It gig-based platform involves phases such as literature review, requirement analysis, system design, software development, and performance evaluation. The focus is on integrating modern technologies to ensure efficiency, reliability, and user convenience. Testing and optimization ensure seamless task management and user engagement.

www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 01, January 2025, pp : 1771-1775

e-ISSN : 2583-1062
Impact Factor : 7.001

3.1 Literature Review & Requirement Analysis

Conducted a comprehensive review of existing gig-based service platforms to analyze their advantages and limitations. Engaged with potential users and workers to gather practical requirements, ensuring that the platform meets expectations for efficiency, security, and ease of use.

3.2 System Design

Designed a scalable architecture using React.js for the frontend and Node.js for the backend. Selected MongoDB as the database for managing user data, tasks, and transactions.

Software Development

Built the platform's core functionalities, including task posting, worker matching, real-time notifications, and secure payments using Stripe. Implemented a leaderboard system and incentives to promote worker engagement.

3.3 Mobile App Development

Developed a responsive mobile interface using React and Tailwind CSS, ensuring a seamless user experience. Integrated push notifications for real-time task updates and worker availability.

3.4 Testing

Conducted rigorous testing to validate backend logic, payment security, and user interactions. Collected user feedback to refine features and improve overall usability.

3.6 Deployment

Deployed the system on a cloud-based server, ensuring scalability and high availability. Verified system performance under real-world conditions to ensure smooth task execution. **2.5 Evaluation & Optimization**

Monitored platform performance using key metrics such as task completion rate, response time, and user engagement. Collected feedback from users and workers, implementing necessary optimizations to enhance efficiency, reliability, and overall user experience.

4. PROJECT OVERVIEW

The clean-It gig-based platform is an innovative solution designed to enhance the efficiency and convenience of on-demand task management. It integrates cutting-edge technologies to connect users with freelance workers for tasks such as waste collection, purchasing goods, and other services. The platform leverages modern web and mobile technologies, including react.js, Node.js, and secure payment systems like Stripe, to provide seamless service delivery. By addressing the limitations of traditional service platforms, clean-It ensures that users can easily post tasks, connect with skilled workers, and make secure transactions, all through a user-friendly interface. This project offers a scalable, practical, and accessible solution for improving service delivery while offering flexible earning opportunities for workers.

System Overview

The clean-It platform utilizes advanced technologies to provide an efficient solution for task management. Users can post tasks through a mobile or web application. The system matches users with appropriate workers based on location, skills, and availability. Payments are securely processed using Stripe, ensuring safe transactions between users and workers. Additionally, features like leaderboards and challenges promote community engagement and sustainability. This compact and efficient system offers a reliable, accessible, and scalable solution for modern gig-based economies.

Key Features

- **Task Management:** Enables users to post, track, and complete tasks seamlessly.
- **JWT Authentication:** Ensures secure user authentication and data integrity through JWT tokens.
- **Mobile & Web Interface**: Provides an intuitive and responsive interface for both users and workers.
- **Secure Payments**: Facilitates seamless and secure transactions using Stripe.
- **Community Engagement**: Incorporates leaderboards and challenges to motivate users and workers.
- **Sustainability**: Promotes eco-friendly practices through partnerships with recyclers.
- **Scalable Solution**: Designed to grow and accommodate increasing user demand.

5. ARCHITECTURE

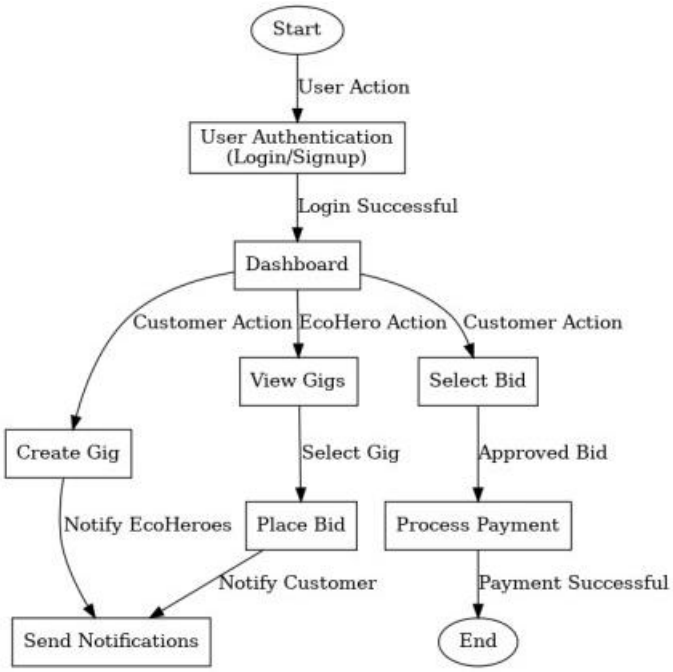
The architecture of the Clean-It project integrates both hardware and software components, offering a seamless gig-based solution where customers can post tasks (gigs) and workers can complete them. Built with a frontend using Next.js and Tailwind CSS, the system provides a responsive and user-friendly experience for task posting, gig worker

www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 01, January 2025, pp : 1771-1775

e-ISSN : 2583-1062
Impact Factor : 7.001

booking, and payment processing. The backend, powered by Node.js, manages API requests, user authentication, task management, and notifications. Data is stored securely using either relational MongoDB databases. AI verification is incorporated for task completion, such as confirming tasks visually or through location data. Secure payments are processed through Stripe, and the platform is hosted on AWS or Google Cloud for scalability and high availability. The system's data flow involves users posting tasks, gig workers accepting tasks, user verifying task completion, payment processing, and updates to leaderboards and ratings. The project ensures efficiency through real-time task verification, increased user engagement with community features, and scalable performance through cloud hosting. It also emphasizes sustainability by promoting eco-friendly practices. The Clean-It system delivers a flexible, secure, and user-friendly solution, making it a cost-effective and scalable choice for managing on-demand services. Architecture Diagram:



6. DESIGN

The design of the Clean-It system integrates advanced technologies to enhance task management and user convenience. The platform utilizes a frontend built with react.js and Tailwind CSS, providing a responsive and user-friendly interface for posting tasks, booking gig workers, and making payments. The backend, developed with React.js, handles API requests, user authentication, task management, and notifications, ensuring seamless communication between the user and the platform. The system also incorporates verification for task completion, such as confirming visual data or location-based tasks, ensuring the accuracy and reliability of completed tasks. Stripe payment integration ensures secure transactions between users and gig workers, while cloud hosting on AWS or Google Cloud ensures scalability and high availability. This modular design ensures real-time communication, efficient task management, and a seamless user experience, offering a cost-effective and scalable solution for on-demand services, with a strong emphasis on sustainability and eco-friendly practices.

7. INTEGRATION

The integration process for the Clean-It system involves combining both hardware and software components to ensure seamless operation. The frontend, built with react.js and Tailwind CSS, interfaces with the backend, which is powered by Node.js for handling API requests, user authentication, and task management. This integration allows for real-time task tracking, notifications, secure payments, and a user-friendly experience, providing a comprehensive solution for on-demand services.

8. TESTING

To ensure the Clean-It system operates as intended, comprehensive testing was conducted:

- **Unit Testing:** Individual components, including the frontend (React), backend (Node.js), and database (MongoDB), were tested to ensure each functioned correctly. This involved validating API endpoints, task management features, and payment processing workflows.

www.ijprems.com
editor@ijprems.com

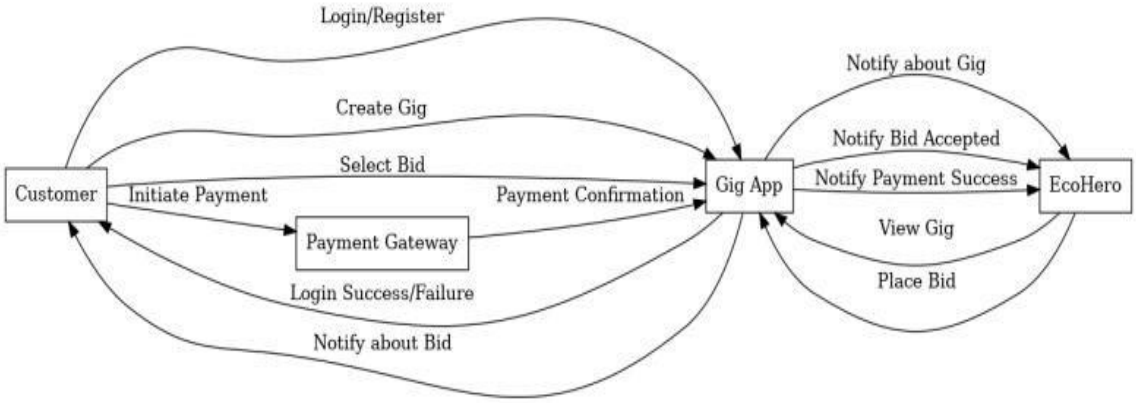
INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 01, January 2025, pp : 1771-1775

e-ISSN : 2583-1062
Impact Factor : 7.001

- **Integration Testing:** The interaction between frontend and backend was verified, confirming that React successfully communicated with the Node.js backend via RESTful APIs. Data retrieval and storage from MongoDB were also tested for accuracy. Stripe payment integration was tested to ensure secure and smooth transaction processing between users and gig workers.
- **Functional Testing:** Functional testing focused on verifying the core features of the system. Task posting and gig worker booking were tested to ensure users could post tasks and workers could accept them. Verification for task completion was tested to ensure it accurately verified tasks through machine learning models or image recognition. Payment processing with Stripe was also tested to confirm secure transactions. All features performed as expected, validating the system's functionality.
- **Performance Testing:** Performance testing focused on evaluating the system's efficiency in handling user requests, task management, and payment processing. The response time for task posting, gig booking, and payment processing was tested to ensure minimal latency. Additionally, the system's scalability was tested by simulating increased traffic, confirming that the platform hosted on AWS or Google Cloud could handle higher user demands effectively. The system demonstrated fast response times, secure transactions, and smooth performance under varying loads, confirming its effectiveness in real-world use.

9. IMPLEMENTATION

The implementation of the Clean-It system involved integrating various components for efficient task management and user interaction. The frontend, developed using React, provides a responsive and user-friendly interface, allowing users to post tasks, book gig workers, track progress, and make payments. The backend, powered by Node.js, handles API requests, user authentication, task management, and notifications. MongoDB is used as the database to securely store user data, task details, and transaction information. The platform leverages Stripe for secure payment processing, ensuring smooth transactions between users and gig workers. Cloud hosting on AWS or Google Cloud ensures scalability and high availability, allowing the system to handle increased traffic and data. The platform was thoroughly tested for functionality, including task posting, worker booking, payment processing, and database interactions, ensuring the system operates efficiently. The design promotes ease of use, scalability, and security, resulting in a flexible and cost-effective solution for on-demand services. Implementation and Design



10. CONCLUSION

The "Clean-It" platform provides an innovative and efficient solution for connecting customers with gig workers for various tasks. By integrating features like task posting, worker assignment, AI verification for task completion, and secure payment processing, the system offers enhanced convenience and reliability for both users and workers.

The successful implementation of the MERN stack architecture, including secure authentication with JWT tokens, and the seamless integration of Stripe for payments, highlights the platform's scalability and robustness. The system's deployment on AWS or Google Cloud ensures reliability and accessibility, making it an ideal solution for individuals and businesses looking for a user-friendly and sustainable approach to managing on-demand services.

11. REFERENCES

[1] Liu Jinping, "Under the IT environment inventory accounting and management studies". Chinese Control and Decision Conference (CCDC) - 29th 2017 IEEE.Ganesh Kumar and P.Vasanth Sena, "Novel Artificial Neural Networks and Logistic Approach for Detecting Credit Card Deceit," International Journal of Computer Science and Network Security, Vol. 15, issue 9, Sep. 2015, pp. 222-234

[2] Vikash K. Singh, Mamata Jenamani, "Composite Web services for Implementing Vendor Managed Inventory".

[3] International Conference on Industrial Informatics IEEE - 5th IEEE, 2007 Chengwei Liu, Yixiang Chan, Syed Hasnain

[4] Alam Kazmi, Hao Fu, "Financial Fraud Detection Fluid: Based on Random Forest," International Journal of Economics and Finance, Vol. 7, Issue. 7, pp. 178-188, 2015.

[5] Friedrich Wiemer, Ralf Zimmermann, "High-speed implementation of bcrypt password search". International Conference on ReConFigurable Computing and FPGAs (ReConFig14), 2014.

[6] M. Weir, S. Aggarwal, B. de Medeiros, B. Glodek, "Password Cracking Using Probabilistic Context-Free Grammars" - IEEE Symposium on Security and Privacy, pp. 391- 405, 2009

[7] Shoham, Yoav, Tennenholtz, Moshe, On social laws for artificial agent societies off line design, Artificial Intelligence, 73(1), 1995: 231-252