International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 03 | March - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Whatsapp Marketing

Mr. J. Noor Ahameed[1], T. KARTHIKAI KUMARAN[2]

[1]Assistant Professor, Department of Computer Applications,

Nehru College of Management, Coimbatore, Tamilnadu, India.

[2]Student of II MCA, Department of Computer Applications,

Nehru College of Management, Coimbatore, Tamilnadu, India.

ABSTRACT:

The **"WhatsApp Marketing System Using API"** is a digital marketing solution designed to streamline and optimize the process of sending promotional messages via WhatsApp. This system is built with a focus on enabling businesses to manage employee recharges, allocate points, and ensure efficient use of messaging resources.

The project consists of three core modules:

1. **Admin Module**: The admin has full control over the system, including managing employee accounts, allocating points based on recharges, and monitoring system activity. Admins can track employee recharges and analyze resource usage.
2. **Employee Module**: Employees can recharge their accounts to earn points, which are used to send messages via WhatsApp. The module allows employees to track their available points, recharge history, and usage logs.
3. **WhatsApp Messaging Integration**: This module integrates with the official WhatsApp Business API or a third-party service to send messages. It facilitates real-time messaging while deducting points for each message sent.

The system is developed using modern web technologies. The frontend is built with **React.js**, offering a dynamic and responsive user interface. The backend leverages **Node.js** and **Express.js**, ensuring robust server-side functionality and efficient API communication. **MongoDB**, a NoSQL database, is used for scalable and flexible data management, storing user recharges, points, and message logs.

By introducing a point-based recharge system, this project provides a transparent and accountable mechanism for message delivery. It enhances operational efficiency with features like user management, message tracking, and system analytics. Integration with the WhatsApp API ensures reliable and real-time communication with customers, enabling businesses to boost engagement and optimize their marketing strategies.

This system serves as an effective and scalable tool for businesses looking to automate and manage their WhatsApp marketing campaigns efficiently.

INTRODUCTION:

In today's digital era, WhatsApp has become one of the most effective platforms for businesses to engage with customers, promote products, and provide services. However, managing large-scale WhatsApp marketing campaigns can be challenging, especially when multiple employees are involved in sending messages. To address this, the **WhatsApp Marketing System Using API** is designed to provide a structured and efficient approach to WhatsApp-based marketing.

This system allows businesses to allocate **message-sending points** to employees based on a **recharge-based model**. Employees can recharge their accounts, earn points, and use them to send messages via WhatsApp. Admins have full control over employee accounts, including monitoring recharges, allocating points, and ensuring proper usage of messaging services.

The project consists of three core modules:

1. **Admin Module** – Admins manage employees, allocate points, and oversee system usage.
2. **Employee Module** – Employees recharge their accounts and send WhatsApp messages using allocated points.
3. **WhatsApp Messaging Integration** – The system connects with a WhatsApp API to facilitate message sending and tracking.

Built using **React.js (frontend), Node.js & Express.js (backend), and MongoDB (database)**, this project ensures a scalable, secure, and user-friendly solution for businesses looking to streamline their WhatsApp marketing efforts. By implementing a point-based system, businesses can **control costs, prevent misuse, and maintain transparency** in their promotional campaigns.

This project aims to **automate, simplify, and enhance WhatsApp-based marketing**, making it a powerful tool for organizations to optimize customer engagement and outreach strategies.

OBJECTIVE:

The primary objective of this project is to develop an efficient, automated, and cost-effective **WhatsApp marketing system** that allows businesses to manage promotional messaging while maintaining control over employee recharges and message usage.

Key Objectives:

1. **Implement a Recharge-Based Messaging System**
   - Enable employees to recharge their accounts to earn points.
   - Ensure that each WhatsApp message sent deducts points from the employee's balance.
2. **Develop a Role-Based Access System**
   - Provide an **Admin Module** to allocate points, manage employees, and monitor system usage.
   - Implement an **Employee Module** where employees can recharge, track points, and send messages.
3. **Seamlessly Integrate WhatsApp Messaging API**
   - Use a third-party API (such as Twilio, Gupshup, or Ultramsg) for sending messages.
   - Enable message tracking, including delivery status and logs.
4. **Ensure Data Security and Scalability**
   - Store user data, recharges, and message logs securely in **MongoDB**.

- Use **Node.js & Express.js** for a scalable backend and **React.js** for a user-friendly frontend.

5. **Provide Transparent Reporting and Analytics** *(Optional)*
   - Allow admins to view reports on employee recharges, points usage, and message history.
   - Generate insights to optimize marketing campaigns.

By achieving these objectives, the system ensures an **organized, controlled, and automated** approach to WhatsApp-based digital marketing, helping businesses optimize engagement while keeping costs in check.

RELATED WORKS:

Several existing WhatsApp marketing tools and automation systems have been developed to help businesses engage with customers efficiently. The **WhatsApp Marketing System Using API** builds upon these existing solutions while introducing a structured recharge-based approach for better control and accountability. Below are some related works that have influenced the development of this system:

1. WhatsApp Business API-Based Marketing Solutions

- Platforms like **Twilio WhatsApp API, Gupshup, and Ultramsg** offer automated WhatsApp messaging services.
- These solutions allow businesses to send notifications, promotional messages, and customer support responses via WhatsApp.
- However, they lack a **recharge-based employee management system**, making it difficult to control usage at an individual level.

2. Bulk WhatsApp Messaging Tools

- Some businesses use bulk messaging tools like **Wati, WhatsApp Cloud API, and AiSensy** to send marketing messages.
- These tools often come with analytics, chatbot integration, and automation features.
- However, they are mostly **subscription-based** and do not provide a **customized points-based system for message allocation**.

3. Recharge-Based Marketing Platforms *(SMS & Email Marketing)*

- Many SMS and email marketing platforms (e.g., Twilio SMS, Mailchimp) follow a prepaid model where users buy credits to send messages.
- This concept is similar to the recharge-based point system in this project but has not been widely implemented for WhatsApp marketing.

4. Custom WhatsApp Automation Using Python & Node.js

- Some developers build custom WhatsApp automation tools using **Node.js (Venom-Bot, OpenWa) or Python (yowsup, WhatsApp Web Automation)**.
- These approaches offer flexibility but **lack structured employee management, tracking, and recharge functionalities**.

METHODOLOGY:

The **WhatsApp Marketing System Using API** follows a structured methodology to ensure efficient message management, controlled employee recharges, and seamless WhatsApp integration. The system is designed with three core modules: **Admin Module, Employee Module, and WhatsApp Messaging Integration**. The development approach involves the following key steps:

1. System Design and Architecture

The system is designed as a **web-based platform** using the **MERN (MongoDB, Express.js, React.js, Node.js) stack**. The architecture consists of:

- **Frontend (React.js)**: Provides an interactive and user-friendly interface for employees and admins.
- **Backend (Node.js & Express.js)**: Handles business logic, user authentication, and API interactions.
- **Database (MongoDB)**: Stores user data, recharges, points, and message logs.
- **WhatsApp API Integration**: Connects with third-party WhatsApp APIs (such as Twilio, Gupshup, or Ultramsg) to send messages.

2. Module Development

A. Admin Module

- **Admin Authentication**: Secure login for administrators.
- **Employee Management**: Add, update, and delete employee accounts.
- **Point Allocation System**: Assign points based on employee recharges.
- **Recharge History & Monitoring**: View transaction history and track employee usage.

B. Employee Module

- **Employee Authentication**: Secure login for employees.
- **Recharge System**: Employees recharge their accounts to earn points.
- **WhatsApp Messaging Interface**: Employees send messages using available points.
- **Message History & Usage Logs**: Employees track their message history and remaining points.

C. WhatsApp Messaging Integration

- **API Connection**: Integration with a third-party WhatsApp Business API.
- **Message Sending Mechanism**: Each message sent deducts points from the employee's balance.
- **Delivery Reports & Logs**: Tracks message status (sent, delivered, failed).

3. Implementation Process

Step 1: Database Design (MongoDB)

- Define **collections** for **users (admins, employees), transactions (recharges), and messages**.
- Establish relationships to track **employee recharges and message usage**.

Step 2: Backend Development (Node.js & Express.js)

- Implement authentication (JWT-based for security).
- Create **APIs** for admin, employee, and WhatsApp messaging functionality.
- Integrate the selected **WhatsApp API** for message delivery.

Step 3: Frontend Development (React.js)

- Design an intuitive **admin dashboard** for managing employees and recharges.
- Develop the **employee panel** for recharges, points tracking, and message sending.
- Implement responsive UI components for a seamless experience.

Step 4: Testing & Debugging

- Perform **unit testing** for backend APIs.
- Conduct **frontend testing** to ensure a smooth user experience.
- Test the **WhatsApp API integration** for accurate message delivery and logging.

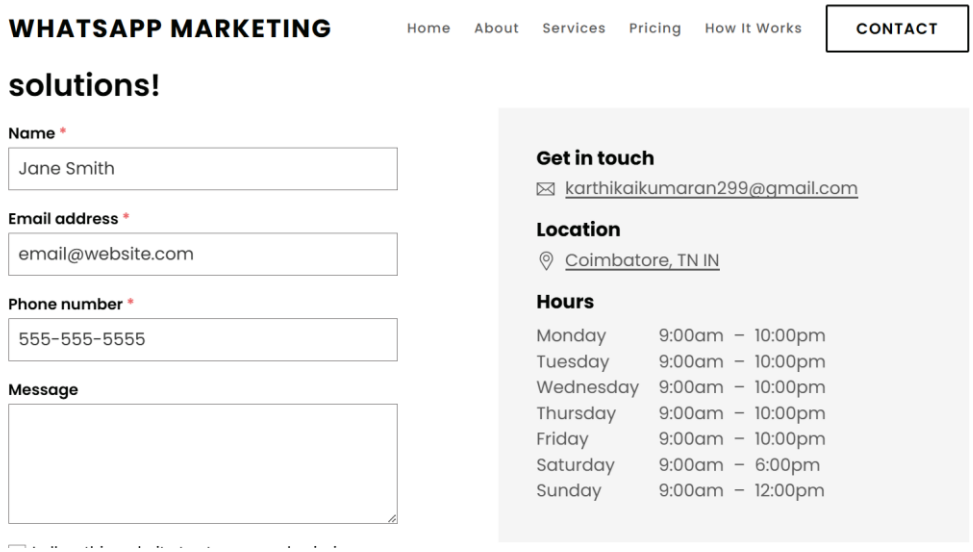
4. Deployment & Maintenance

- Host the application on a cloud server (**AWS, Vercel, or DigitalOcean**).
- Set up **MongoDB Atlas** for database hosting.
- Implement **security measures** (rate-limiting, HTTPS, authentication).
- Provide **regular updates** to enhance performance and security.

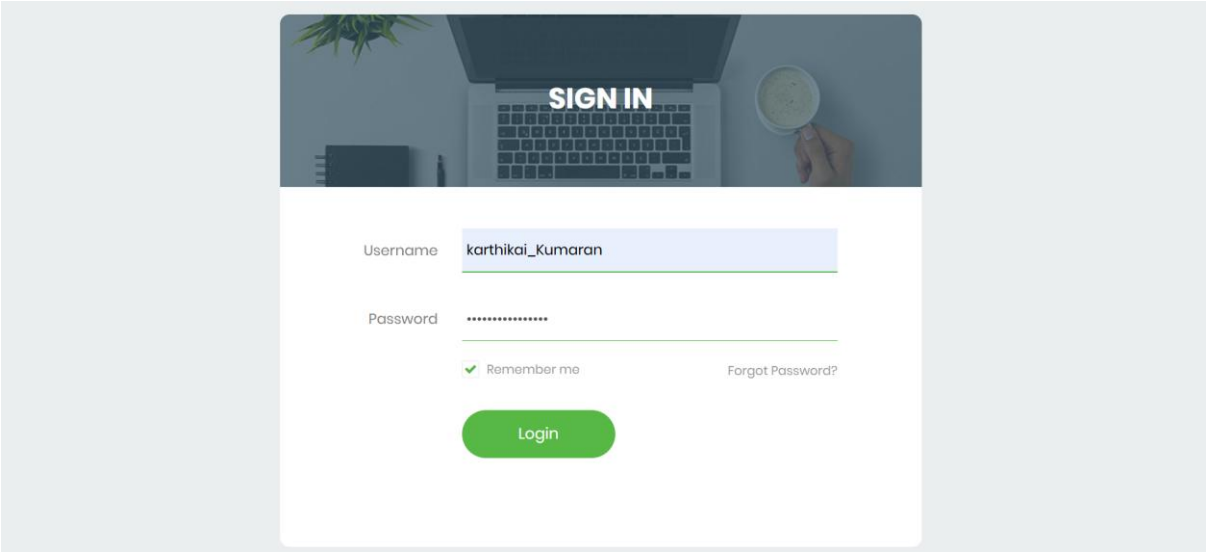
RESULTS:

The **WhatsApp Marketing System Using API** successfully provides a structured and automated approach to managing WhatsApp-based promotional messaging. The implementation of a recharge-based points system ensures that businesses can efficiently control employee message usage while preventing excessive or unauthorized communication.

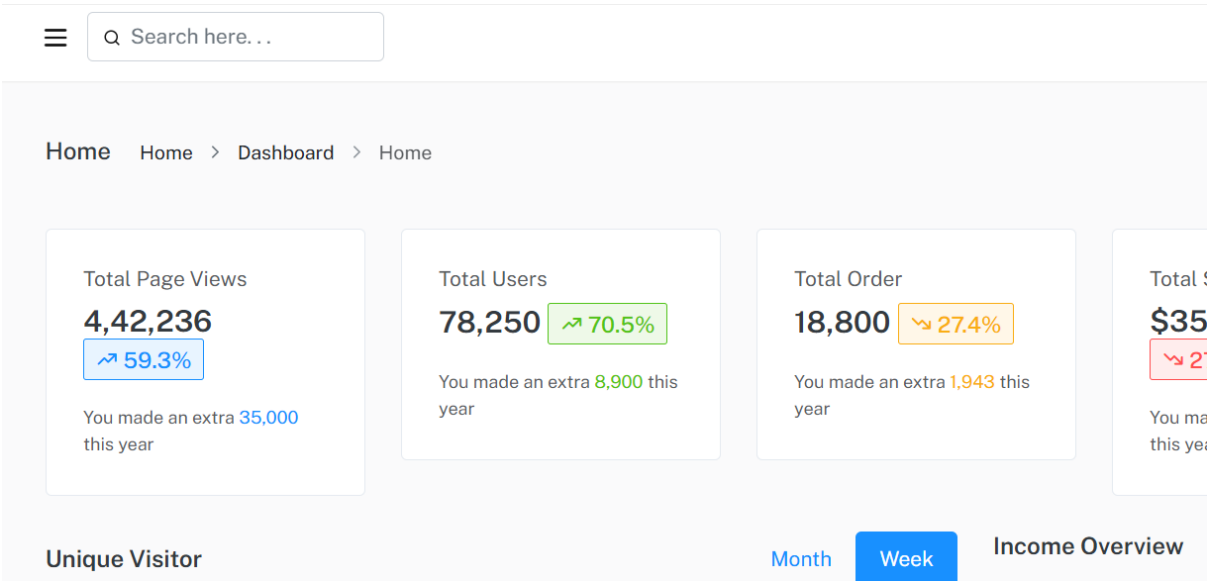
Sample Photos



User Contact Us Page



Admin login Page



Admin dashboard

CONCLUSION:

The **WhatsApp Marketing System Using API** is an efficient, scalable, and structured solution designed to automate and optimize WhatsApp-based promotional messaging. By implementing a **recharge-based message-sending model**, the system ensures controlled usage, cost efficiency, and transparency in digital marketing operations.

Through **three core modules (Admin, Employee, and WhatsApp Messaging Integration)**, businesses can manage employees, allocate points, and track messaging activities effectively. The **React.js (frontend), Node.js & Express.js (backend), and MongoDB (database)** technology stack ensures a modern, responsive, and scalable architecture. Integration with a **WhatsApp API** enables real-time message delivery and tracking, making the system highly functional for business communication.

This project not only simplifies WhatsApp marketing operations but also provides a **secure, user-friendly, and data-driven approach** to digital outreach. By enabling **structured message control and recharge-based usage**, it prevents misuse and optimizes marketing efforts.

In the future, the system can be enhanced with **advanced analytics, AI-powered chatbots, message scheduling, and multi-language support** to further improve automation and engagement.

This project serves as a **valuable tool for businesses looking to leverage WhatsApp marketing efficiently**, ensuring better customer engagement, controlled expenses, and improved campaign management.

REFERENCE:

- ✓ Node.js Foundation. (2020). *Node.js Documentation.* Retrieved from https://nodejs.org/en/docs/.
- ✓ Angular Team. (2020). *Angular Developer Guide*. Google Developers. Retrieved from https://angular.io/docs.
- ✓ Express.js. (2020). *Express.js Documentation.* Retrieved from https://expressjs.com/.

✓ MySQL Documentation. (2021). *MySQL 8.0 Reference Manual*. Oracle Corporation. Retrieved from https://dev.mysql.com/doc/.
✓ Stack Overflow. (2024). *Various Technical Discussions on Angular, Node.js, and MySQL*. Retrieved from https://stackoverflow.com.
✓ ChatGPT. (2024). *Assistance with Project Documentation and Technical Queries*. OpenAI.
✓ Twilio. (2024). Twilio WhatsApp Business API Documentation. Retrieved from https://www.twilio.com/docs/whatsapp.
✓ Gupshup, MongoDB. (2024). NoSQL Database for Scalable Applications. Retrieved from https://www.mongodb.com/docs/manual/.
✓ React.js. (2024). A JavaScript Library for Building User Interfaces. Retrieved from https://react.dev.
✓ Node.js. (2024). JavaScript Runtime for Scalable Network Applications. Retrieved from https://nodejs.org/en/docs.
✓ Express.js. (2024). Fast, Unopinionated Web Framework for Node.js. Retrieved from https://expressjs.com.
✓ WhatsApp API for Business Communication. Retrieved from https://gupshup.io/developer/docs/whatsapp-api